

Reconfigurable Wireless Networks

Amr El-Mougy, Mohamed Ibnkahla, Ghaith Hattab, and Waleed Ejaz

Abstract—Driven by the advent of sophisticated and ubiquitous applications, and the ever-growing need for information, wireless networks are without a doubt steadily evolving into profoundly more complex and dynamic systems. The user demands are progressively rampant, while application requirements continue to expand in both range and diversity. Future wireless networks, therefore, must be equipped with the ability to handle numerous, albeit challenging requirements. Network reconfiguration, considered as a prominent network paradigm, is envisioned to play a key role in leveraging future network performance and considerably advancing current user experiences. This paper presents a comprehensive overview of reconfigurable wireless networks and an in-depth analysis of reconfiguration at all layers of the protocol stack. Such networks characteristically possess the ability to reconfigure and adapt their hardware and software components and architectures, thus enabling flexible delivery of broad services, as well as sustaining robust operation under highly dynamic conditions. The paper offers a unifying framework for research in reconfigurable wireless networks. This should provide the reader with a holistic view of concepts, methods, and strategies in reconfigurable wireless networks. Focus is given to reconfigurable systems in relatively new and emerging research areas such as cognitive radio networks, cross-layer reconfiguration and software-defined networks. In addition, modern networks have to be intelligent and capable of self-organization. Thus, this paper discusses the concept of network intelligence as a means to enable reconfiguration in highly complex and dynamic networks. Key processes in network intelligence, such as reasoning, learning, and context-awareness are presented to illustrate how these methods can take reconfiguration to a new level. Finally, the paper is supported with several examples and case studies showing the tremendous impact of reconfiguration on wireless networks.

Index Terms—Cognitive radio networks, context-awareness, cross-layer reconfiguration, learning, machine-to-machine communications, network intelligence, reconfigurable networks, software-defined networks.

I. INTRODUCTION

The last few years have witnessed an incredible revolution in wireless networks. The number of wireless devices worldwide continues to increase at an enormous rate and the applications of wireless networks are getting increasingly versatile, sophisticated, and ubiquitous. Thus, future wireless networks are expected to operate under several challenging conditions. First, they are expected to provide efficient always-on reliable high data rate access. Second, they have to be energy-efficient for the sake of user convenience and for environmental concerns, since the widespread of wireless devices leads to significant increase in their total energy consumption and greenhouse gas (GHG) emission. Indeed, information and communication systems consume over 10% of the world's total energy [1]. Third, wireless networks have to consider challenging and

changing user demands (video traffic over mobile devices now accounts for over 50% of global data traffic [2]). Fourth, users expect their devices to be *smart*, to learn about their interests, and to provide them with information that is timely and convenient (over 2 billion smartphone subscriptions have become available in 2014 [3]). Last but not least, the spread of cloud resources and big data centres, together with the advent of the Internet of Everything (IoE), promise to provide wireless users with unprecedented services and opportunities.

These challenges have forced designers of wireless systems to completely change their traditional design methodologies. Primarily, the above challenges mean that modern wireless networks have to be able to consider a wide range of changing objects and conditions. Thus, modern wireless systems have to be able to adapt to the evolving nature of wireless networks and services. In other words, they have to be reconfigurable across multiple layers of the protocol stack. For example, modern devices have to adapt their transmission schemes at the physical (PHY) layer in order to operate in time-varying conditions or high mobility, they have to reconfigure their routing protocols to operate in heterogeneous or evolving networks, and they have to adapt to their application requirements in order to satisfy a wide range of user interests. Therefore, reconfiguration will be a key enabler for next-generation wireless systems and services.

We can classify research in reconfigurable networks into three levels: Reconfiguration at the PHY layer, reconfigurable networking, and network intelligence. This hierarchy is shown in Fig. 1 with a few examples of research topics at each level.

Reconfiguration at the PHY layer covers a broad range of topics such as adaptive modulation, antenna beamforming,

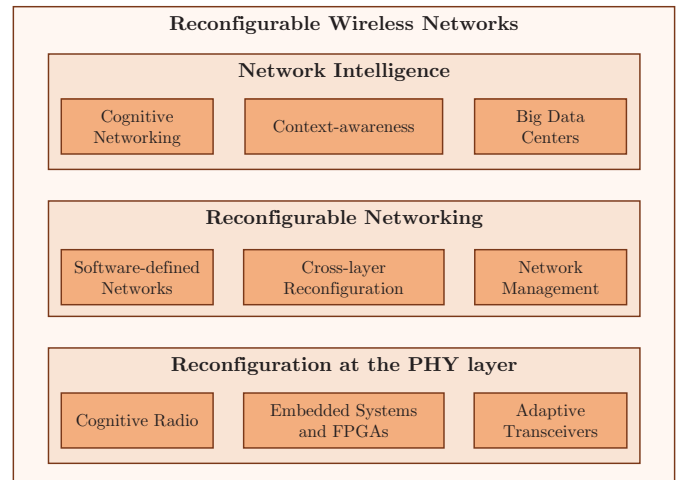


Fig. 1. Classification of research in reconfigurable networks and examples of research topics.

The authors are with the department of Electrical and Computer Engineering at Queen's University, Kingston, ON, Canada (email: {mohamed.ibnkahla; bema; g.hattab; we5}@queensu.ca).

and software-defined radio (SDR). Work in these functions has been covered comprehensively in the literature [4]–[12]. Section II focuses on reconfiguration at the PHY layer, where network-wide considerations are taken into account during device reconfiguration. With the recent emergence of cognitive radio networks (CRNs), and the significant role CRNs are expected to play in next-generation wireless networks, a special discussion on CRNs is given in Section II.

Reconfigurable networking is concerned with building adaptive networking firmware that can address changing network topologies or application requirements. Thus, Section III studies methodologies of implementing reconfiguration at the networking level. Three particular aspects of reconfigurable networking are considered: Cross-layer design, network management, and software-defined networks (SDN). Several examples are thoroughly analyzed in order to demonstrate how reconfiguration can solve different network challenges. Performance analysis is also presented to show the gain achieved from reconfiguration. For illustration purposes, this paper considers the layered model presented in Table I. The model is composed of the application, transport, network, medium access control (MAC), and PHY layers. Cross-layer reconfiguration is the leading methodology in this area because it has demonstrated significant improvements in the network performance when inter-layer effects are taken into consideration [4], [13]. For example, PHY layer parameters such as transmit power and bit error rate (BER) can have profound effects on several processes in other layers of the protocol stack such as scheduling at MAC layer and topology control at the network layer. Therefore, the single-layer design approach can tangibly limit the adaptations that can be implemented. In addition, network management is an important tool for reconfiguring network parameters and protocols from a network-wide perspective, which provides the potential to consider global network objectives. On the other hand, SDN is a promising method that can leverage capabilities in automation and virtualization of networks.

The third level of reconfiguration in wireless networks

TABLE I
RECONFIGURATION CAN BE PERFORMED AT ALL LAYERS

Layers	Examples of Possible Reconfigurations
Application	Context information
	User requirements
	User interface
Transport	Number of retransmissions
	Congestion control
Network	Routing
	Admission control
	QoS management
MAC	Transmission and sleep scheduling
	Contention/sensing window
	Transmission rate
PHY	Transceiver and antenna reconfiguration

adopts intelligence and cognitive strategies throughout the wireless system and its surrounding environment [14]. Intelligence and cognition include reasoning and learning, which aim at improving networking decisions while fulfilling network end-to-end goals. Of particular importance in this level is context-awareness (CA) [15], which enables networking decisions to be made according to certain relevant information. Here, the goal is to reconfigure the network to suit system-wide architecture/deployment, user interests, or the environment, often through predictions and proactive actions. This level is discussed in Section IV.

It is clear from this introduction that research in reconfigurable wireless networks has quite a wide scope. It also demands expertise from multiple disciplines, mainly engineering and computer science. However, publications in this field have been quite scattered, often focusing on a limited set of issues. In this paper, we provide a unifying framework for research in reconfigurable wireless networks. The objective is to link together ideas from different fields and provide the reader with a holistic view of concepts, methods, and strategies in reconfigurable wireless networks.

The rest of the paper is organized as follows. Section II covers the PHY layer reconfiguration from a network perspective, focusing on CRNs. A case study is provided to examine the performance gains achieved when network-wide reconfiguration is implemented. Section III covers reconfigurable networking, with emphasis on: Cross-layer reconfiguration, network management, and SDNs. A case study is presented to show the performance gains archived when cross-layer reconfiguration is used in machine-to-machine (M2M) networks. Section IV is devoted to cognitive networking and context-awareness. A case study is presented which investigates network management using reasoning and learning in wireless sensor networks. Finally, the main conclusions and open research areas are presented in Section V.

II. PHY LAYER RECONFIGURATION: A NETWORK PERSPECTIVE

PHY layer reconfiguration includes SDR, transceiver reconfiguration, adaptive antennas and beamforming, adaptive modulation and coding (AMC), etc. [5], [6], [16]. Most publications in this field target device-level reconfiguration that often does not take into consideration the network aspect (even when wireless devices are part of a wider network). This section shows the importance of network-based device reconfiguration. Without loss of generality, and in order to illustrate the different concepts to the reader, this section explores the case of CRNs.

CRNs are envisioned to tackle the problem of low utilization of the spectrum by exploiting spectral opportunities in time, frequency, and space. These opportunities can be used by secondary users (SUs), as illustrated in Fig. 2, in situations where the incumbents (also known as primary users (PUs)) of these resources are idle, or they can be shared by both PUs and SUs if the latter conform to stringent rules to protect the former from harmful interference.

Conventionally, a wireless network is primarily concerned with providing some services at certain performance levels

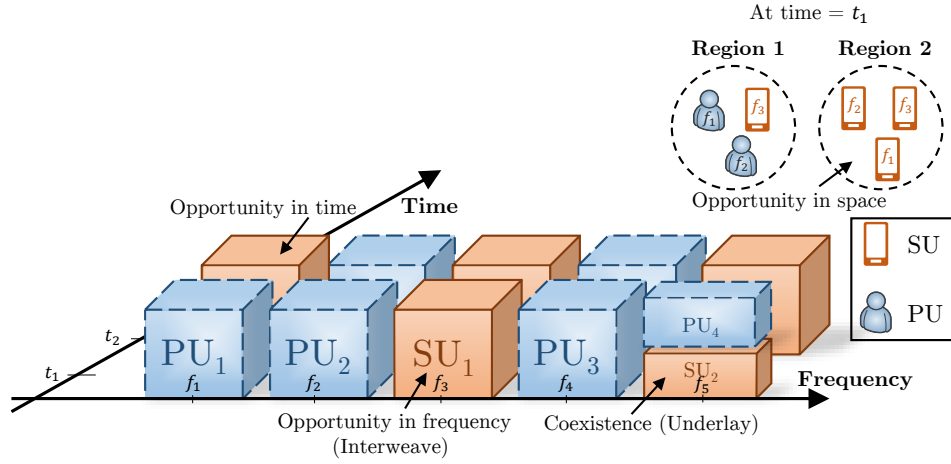


Fig. 2. There are unused spectral opportunities in time, frequency, and space. SUs can orthogonally exist with PUs (interweave paradigm), or coexist with them and share the resources (underlay paradigm).

to its users (e.g., throughput, fairness, etc.). However, the CRN must provide guarantees to both SUs and PUs. In particular, the CRN must not degrade the quality-of-service (QoS) of PUs. It must also provide QoS guarantees to SUs. The coexistence of CRNs with heterogenous PU networks (PUNs) is challenging because:

- Different PUNs may have different interference tolerance levels.
- Some PUs are static (e.g., TV channels) while others are mobile (e.g., cellular users). Protecting the latter is more challenging.
- Some PUs transmit at low powers (e.g., wireless microphones), and thus detecting the presence of these incumbents is challenging.

In addition, providing a certain QoS for SUs remains a challenging task mainly because:

- The spectrum access is opportunistic. Hence, the resources are limited, and they must be shared among multiple SUs efficiently and in a fair manner.
- The PU activity may change. Thus, periodic monitoring of the spectral opportunity is vital, meaning that data transmission interruptions or spectrum handover to other available opportunities are inevitable.

To deal with these challenges, network level *reconfigurability* and *learning* are key enablers for effective CRNs.

In the subsequent subsections, we discuss the key functionalities of a CRN, and particularly, the cognitive management module, which acts as the brain of the CRN. Then, we examine different levels of reconfiguration and evaluate the throughput performance of a CRN in the absence and presence of network-based reconfiguration and learning. We show that significant gains can be achieved with proper reconfiguration at the SU device and the network level.

A. Cognitive Management

The CRN must be equipped with a cognitive management module, which is composed of a *cognitive spectrum manager* and a *cognitive resource manager* as shown in Fig. 3 [17],

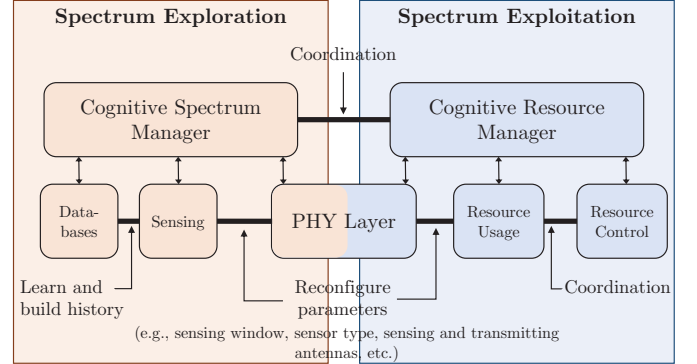


Fig. 3. Spectrum exploration and exploitation must work in tandem to utilize the spectral opportunities.

[18]. The former helps *explore* the spectral opportunities, and the latter helps *exploit* spectral resources and allocates them to different SUs.

1) *The cognitive spectrum manager (CSM)*: It is responsible for exploring spectral opportunities over the multi-dimensional domain (time, frequency, and space). This can be realized through accurate databases (e.g., databases of the location of TV channels), spectrum sensing (e.g., energy detection), or a cooperation of both. In this section, multiband sensing is assumed because it identifies a wider range of spectral opportunities compared to single-band techniques [19]. However, instead of exploring the entire spectrum by a single SU, multiple SUs can cooperate to efficiently identify the available channels, where those with the highest potential are explored first [19].

2) *The cognitive resource manager (CRM)*: It has several key responsibilities. First, it collaborates with the CSM to:

- Divide the multiband spectrum sensing task among different SUs, based on different possible factors (e.g., the SU's capability, battery level, location, etc.).
- Sense specific parts of the spectrum instead of sensing the entire spectrum. For instance, to guarantee a certain level of QoS for SUs, the channels with the highest signal-to-

noise ratio (SNR) as well as the channels with the lowest PU activity are sensed first.

Second, once the CSM identifies the spectral opportunities for current and future use, the CRM must divide these resources among the different SUs. This is realized through *resource usage* (RU) and *resource control* (RC) modules [17]. The RU module is responsible for reconfiguring the PHY layer to exploit the resources (e.g., spectrum shaping, which can be done via orthogonal frequency division multiplexing (OFDM)). The RC module is responsible for resource allocation (and reallocation in the event of returning PUs) and mobility management (which includes spectrum handover).

B. Cognitive Radio Layers Reconfiguration

Intuitively, the CSM and the CRM must work in tandem and provide cross-layer optimization to guarantee QoS support for SUs and to protect PUs. Network reconfiguration is not solely used at the PHY layer, but across all layers, from the application layer down to the PHY layer as illustrated in Table I. For instance, the application layer determines the QoS requirements for the SUs (e.g., different applications have different bandwidth requirements, data rates, delay-tolerance, etc.). Therefore, the wideband receiver front-end is reconfigured to support the QoS requirement of a given application at the SU's device level. At the network level, the number of sensing SUs can be reconfigured to meet these requirements (e.g., more SUs can be asked to sense a channel that requires very high sensitivity for detecting a PU). At the transport layer, delays and SU data packet losses may not only be induced due to the channel, mobility, or congestion, but may also be due to spectrum handover (e.g., when a PU reclaims a channel), or when an intermediate SU (acting as a hop) cannot properly relay information [20], [21]. To overcome these unique challenges, the CRM must identify a list of candidate and backup spectral opportunities.

At the network layer, cognitive managers must learn and recognize the traffic patterns of the PUNs (also known as *network tomography* [22]). This is necessary because it helps the network in routing reconfiguration and in understanding the interference and spectrum usage patterns related to PUs. Moreover, routing in CRNs must be spectrum-aware in order to consider the dynamic nature of the available spectral opportunities. For example, routing protocols must not only consider the shortest paths (to minimize energy) and the strongest links (to maximize throughput), but also paths that generate the minimal interference to PUs [22], [23].

At the MAC layer, access protocols must be intelligently aware of the spectrum. Since the resources are dynamically changing, access control must be flexible and must consider practical constraints such as the availability of a common control channel for SUs to share the sensing results. In addition, the MAC layer directly controls the reconfiguration of the SU's transceiver. For instance, reconfiguring the sensing schedule by distributing the task of multiband sensing over multiple SUs is essential in order to reduce the high complexity of multiband sensing. Moreover, the MAC frame includes a sensing slot and a transmission slot. Thus, an inherent-tradeoff between

sensing accuracy and network throughput must be optimized by properly reconfiguring the sensing time to maintain a balance between QoS delivery for SUs and protection for PUs [24].

At the PHY layer, several reconfiguration strategies can be adopted for both sensing and transmission:

- The SU can reconfigure the spectrum sensor based on channel conditions, prior channel state-information (CSI), prior knowledge about the PU signal, or any combination of these. For example, energy sensors can be used in the absence of any prior knowledge about the PU signal and when the channel condition is good. Coherent and feature-based sensors can be used if the SU has prior information about the PU signal [19], [23], [25].
- In the case of multiband sensing, spectrum sensing can be reconfigured to sense a subset of channels using, for instance, filter banks [26].
- The parameters of the spectrum sensor can be dynamically reconfigured (e.g., the number of collected samples, the decision threshold, etc.).
- In the case of multiple-input multiple-output (MIMO) SUs, antennas can be reconfigured into transmitting antennas, sensing antennas, or both. In the last case, concurrent sensing and transmission is feasible, under certain conditions [27].
- For transmission, multiple SUs may access non-contiguous channels, and thus spectrum sculpting is essential to circumvent interfering with PUs (see [28] for more details about spectrum sculpting techniques).

Besides all the aforementioned hardware reconfigurations, conventional reconfiguration can also be used such as power control and AMC schemes.

C. Case Study

In this section we study a CRN where network-based reconfiguration is envisioned and compared with a basic CRN (in the absence of network reconfiguration).

We consider a network that consists of K SUs $\in \mathcal{K} = \{SU_1, SU_2, \dots, SU_K\}$, and M channels at center frequencies $\in \mathcal{F} = \{f_1, f_2, \dots, f_M\}$. The PU activity (i.e., the probability that the PU is active) for each channel, $0 \leq p_m \leq 1$, $m = 1, 2, \dots, M$, is assumed to be known (it can be obtained using PU traffic estimation techniques [29]). For instance, $p_m = 0.3$ means that the m -th channel is occupied 30% of the time.

Each SU senses multiple channels to determine whether they are occupied or not. That is, SU_k solves the following multiple binary hypothesis testing problem [19]

$$\begin{aligned} \mathcal{H}_0^{m,k} : \mathbf{y}_m^k &= \mathbf{w}_m^k \\ \mathcal{H}_1^{m,k} : \mathbf{y}_m^k &= \mathbf{x}_m + \mathbf{w}_m^k, \end{aligned} \quad (1)$$

where $m = 1, 2, \dots, I_k \leq M$ is the channel index, such that SU_k senses I_k channels, $\mathbf{y}_m^k = [y_{m,1}^k, y_{m,2}^k, \dots, y_{m,N_m^k}^k]^T$ is the sensed signal at the m -th channel by SU_k , \mathbf{x}_m is the transmitted PU signal, and \mathbf{w}_m^k is a zero-mean additive white Gaussian noise (AWGN) with unit variance. SU_k decides $\mathcal{H}_0^{m,k}$ if the m -th channel is unoccupied or decides $\mathcal{H}_1^{m,k}$ otherwise.

In general, SU_k forms a test statistic based on the likelihood ratio test. Thus, the decision rule of SU_k for the m -th channel is

$$\Lambda(\mathbf{y}_m^k) \underset{\mathcal{H}_0^{m,k}}{\overset{\mathcal{H}_1^{m,k}}{\gtrless}} \xi_m^k, \quad (2)$$

where $\Lambda(\mathbf{y}_m^k)$ is the test statistic, and ξ_m^k is the threshold that divides the decision region into $\mathcal{H}_1^{m,k}$ and $\mathcal{H}_0^{m,k}$. To form a test statistic for the m -th channel, SU_k collects N_m^k samples. This parameter is commonly known as the *sensing window*. It plays a key role in the sensing-throughput tradeoff [24].

We consider three spectrum sensors: An energy-based sensor and two feature-based sensors (FBS). The first FBS exploits pilot patterns in the frequency spectrum, and the second exploits the cyclic prefix in an OFDM-based PU. The reader may refer to [19], [23], [25] for a comprehensive discussion of state-of-the-art sensing techniques.

If the SU has no prior knowledge about the PU signal, or if the channel link is good, then the SU may use the energy detector (ED), which is expressed as

$$\frac{1}{N_m^k} \|\mathbf{y}_m^k\|^2 \underset{\mathcal{H}_0^{m,k}}{\overset{\mathcal{H}_1^{m,k}}{\gtrless}} \xi_{m,ED}^k, \quad (3)$$

where $\|\cdot\|^2$ is the Frobenius norm. This spectrum sensor has the following receiver operating characteristic (ROC) performance under the AWGN channel condition [24]

$$P_{D_{k,m}}^{ED} = \mathcal{Q}\left(\frac{\mathcal{Q}^{-1}(P_{FA}) - \sqrt{N_m^k \text{SNR}_k}}{\sqrt{2\text{SNR}_k + 1}}\right), \quad (4)$$

where $P_{D_{k,m}}^{ED}$ is the ED detection probability of SU_k at the m -th channel, P_{FA} is the false alarm probability requirement, SNR_k is the SNR at SU_k , $\mathcal{Q}(\cdot)$ is the complementary distribution function of the standard Gaussian, and $\mathcal{Q}^{-1}(\cdot)$ is its inverse.

In many practical systems, the PU transmits a pilot pattern along with the data-carrying signal. If the SU has a prior knowledge about the pilot pattern, \mathbf{x}_m^P , then it may use the pilot detector (PD), which is expressed as

$$\frac{1}{N_k} (\mathbf{x}_m^P)^H \mathbf{y}_m^k \underset{\mathcal{H}_0^{m,k}}{\overset{\mathcal{H}_1^{m,k}}{\gtrless}} \xi_{m,PD}^k, \quad (5)$$

where $(\cdot)^H$ is the Hermitian operator. The ROC performance of this sensor (in the AWGN channel case) is expressed as [30], [31]

$$P_{D_{k,m}}^{PD} = \mathcal{Q}\left(\mathcal{Q}^{-1}(P_{FA}) - \sqrt{2\theta N_m^k \text{SNR}_k}\right), \quad (6)$$

where θ is the power allocation factor for the pilot signal.

Finally, using the detector proposed in [32] for detecting OFDM-based PUs, the ROC performance can be approximated by

$$P_{D_{k,m}}^{OFDM} \approx \mathcal{Q}\left(\frac{\mathcal{Q}^{-1}(P_{FA}) - \sqrt{2N_m^k \text{SNR}_k}}{\sqrt{4\text{SNR}_k + 1}}\right). \quad (7)$$

Fig. 4 illustrates the detection performance of the three spectrum sensors in the single SU single-band case. The sensing duration is $N = 1000$ samples for all sensors to achieve a

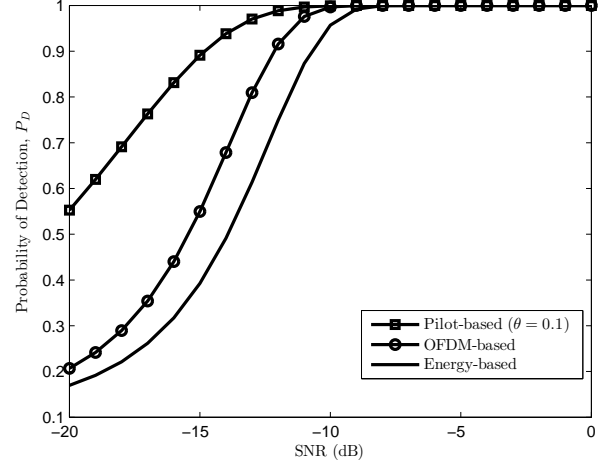


Fig. 4. Detection performance of the three spectrum sensors.

fixed false alarm probability of 10%. It is observed that the pilot-based detector achieves the best detection performance at very low SNR, whereas the energy detector exhibits a poor performance because it does not incorporate any prior knowledge about the PU signal.

We present now a number of reconfiguration strategies for the CRN:

1) *Basic CRN*: This is the simplest scenario where the network arbitrarily allocates the sensing tasks to the SUs. Here, each individual SU merely configures its transmission frequency according to the spectrum availability. Particularly, in this scenario:

- Each SU may have to sense all channels in the absence of any coordination. Thus, $I_k = M \forall k = 1, 2, \dots, K$ regardless of the PU activity profile. Note that this implies that the SU may sense a channel with $p_m \approx 1$ (i.e., almost always busy), which is clearly a waste of resources. Moreover, sensing the entire spectrum by an individual SU is expensive and complex.
- All SUs sense for a fixed duration (i.e., $N_m^k = N \forall k = 1, 2, \dots, K, \forall m = 1, 2, \dots, M$). That is, the SU does not reconfigure its sensors based on the channel quality. This may lead to throughput losses especially if a channel is in a good condition and the sensing window is not reduced for that channel.
- Each SU implements a *non-reconfigurable multiband detector* (NMD), as shown in Fig. 5(a). It consists, in this case, of M EDs. Here, the same threshold is used at SU_k , i.e., $\xi_{m,ED}^k = \xi_{ED}^k$, regardless of PU activities and interference protection levels.

Once the sensing results are obtained for all channels, the SUs send their local decisions to a fusion center (FC) to make a global decision on the occupancy of the M channels.

2) *CRN with device-based reconfiguration*: Here, each SU optimizes the thresholds and the sensing duration of each branch of its multiband detector to adapt to the interference levels, channel quality, etc. For instance, the *multiband joint detector* (MJD), which consists of energy sensors, jointly

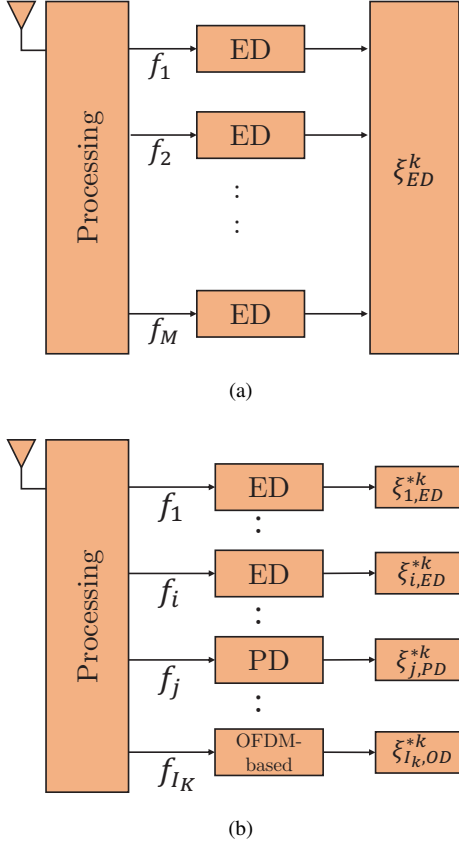


Fig. 5. (a) The NMD consists of multiple single-type detectors (in this case, all detectors are EDs); (b) The RMD may consist of multiple detectors of multiple types (in this case, there are EDs, PDs, and OFDM-based detectors).

optimizes $\xi_{m,ED}^k$ and N_m^k (see [33]–[35] for more details) to maximize the CRN throughput. A higher level of reconfiguration is obtained when the multiband detector consists of sensors of various types, as illustrated in Fig. 5(b). We refer to it as the *reconfigurable multiband detector* (RMD).

3) CRN with centralized network-based reconfiguration:

Here, the FC reconfigures the network's parameters and coordinates the spectral sensing and exploitation tasks of SUs. In particular, the FC may group the SUs into multiple clusters. Each cluster is responsible of sensing a subset of the wideband spectrum. Furthermore, the FC can assign the sensing tasks among the SUs based on various factors (e.g., the FC can ask an SU to only sense the channels with the highest SNR at its location).

4) CRN with distributed network-based reconfiguration:

The centralized CRN has several limitations. First, the amount of information exchange and network overhead between the SUs and the FC is tremendous due to continuous spectrum sensing and decision updates performed by the SUs in real-time. Second, in practice the communication links between the SUs and the FC are not error-free. Indeed, the network's performance can be limited if the reporting channel is in a deep fade [36]. Third, many network architectures lack a common central unit (e.g., ad hoc networks). For these reasons, distributed network-based reconfiguration is appealing for CRNs. Due to the absence of the FC, *learning* becomes prominent in

order to improve the reliability of spectrum sensing [23], [37]–[41]. Here, a consensus-based learning scheme is considered. It consists of two stages: In the first stage, each SU senses its corresponding list of channels using either the NMD or the RMD (depending which device reconfiguration is in effect). In the second stage, each SU communicates with its neighboring SUs for information exchange. Each SU_k checks with its neighboring SUs, and if some of them are sensing some channels that SU_k is sensing, then it will use the information obtained from these neighbours to update its own test statistics as follows

$$\Lambda(\mathbf{y}_m^k(i+1)) = \Lambda(\mathbf{y}_m^k(i)) + \mu \sum_{l \in \mathcal{N}_k} \Lambda(\mathbf{y}_m^l(i)) - \Lambda(\mathbf{y}_m^k(i)) \quad (8)$$

where i is the iteration index, \mathcal{N}_k is a set of SU_k 's neighbours, and μ is the learning step size.

For example, let $\Lambda(\mathbf{y}_m^k) = E(k, m)$ be the energy of the samples collected by SU_k for the m -th channel. Then, SU_k will exchange information about this energy measurement with its neighbors until all corresponding SUs reach a consensus about the energy level of that channel (i.e., until $E(k, m)$ converges to some constant $\forall k \in \mathcal{N}_k$). See [41] for an in-depth analysis of the convergence properties of consensus-based algorithms.

5) *Performance analysis:* The average aggregate throughput of the CRN is investigated for different levels of reconfiguration and compared to the basic CRN (see [19] for more details about the throughput in multiband CRNs). The case of $M = 5$ channels and $K = 10$ users is considered. Each user senses $I_k = I = 2 \forall k$ channels (the same assumption is made for the basic CRN for a fair comparison) and transmits its own data over one available channel with transmit power of 10dB. The SNR of each channel at each SU is based on a Rayleigh fading model, with an average SNR of $\bar{\gamma}$. Moreover, the PU activity over the m -th channel is assumed to follow a uniform random process, i.e., $p_m \sim \mathcal{U}(0, 1)$, and it is independent of the other channels.

In the centralized case (Fig. 6(a)), SUs send their decisions to the FC, which combines the results using OR-logic rule [42]. The resulting decisions are then sent by the FC to the SUs for spectrum access.

For the basic CRN scenario, each SU senses two pre-assigned channels using the ED-based NMD (i.e., it is assumed that the FC allocates two channels for each SU such that each channel is sensed by $(I \times K)/M$ SUs as shown in Fig. 6(a)).

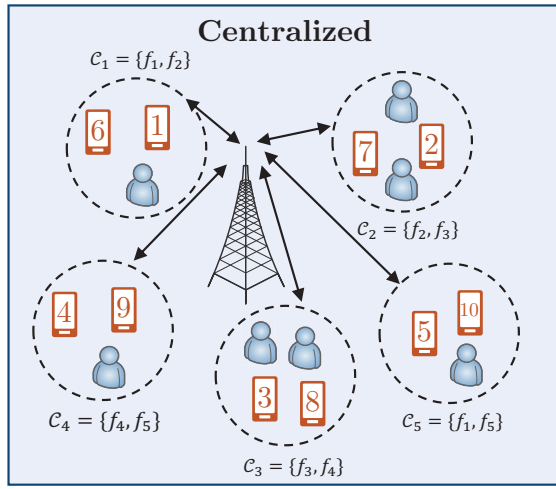
For the CRN with device-based reconfiguration, the channel allocation is similar to that of the basic CRN case. However, the SUs jointly optimize the sensing duration and the threshold of each channel (to maximize the throughput over that channel subject to a false alarm constraint of 10%). Here, two cases are studied: In the first one, the SUs use the MJD, and in the second case they use the RMD. Without loss of generality, let us assume that an SU with the RMD uses a PD for f_1 and f_2 , an ED for f_3 and f_4 , and an OFDM-based detector for f_5 .

For the CRN with network-based reconfiguration (centralized and distributed), the sensing tasks are no longer randomly allocated. In this study, each SU senses the channels that have the highest SNRs at their locations.

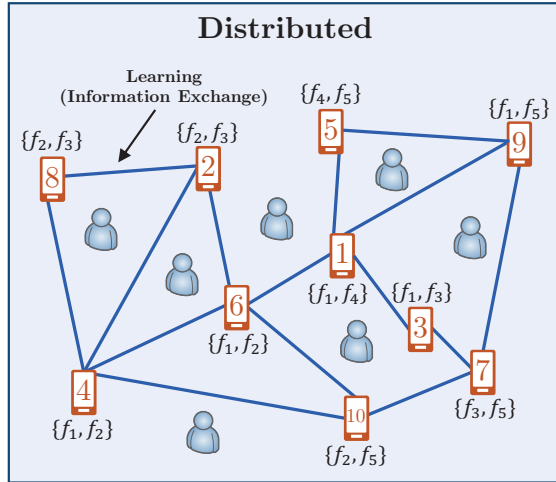
For the distributed network-based reconfiguration case (Fig. 6(b)), it is assumed that each SU is aware of its neighbours. In addition, each SU collects a fixed number of samples, N , for the sensing process. The duration of the learning process is set to 10 iterations, with a step size of $\mu = 0.25$. It is reasonable to use a fixed duration for the sensing and learning processes because synchronization among SUs is essential in distributed networks.

The aggregate throughput of the centralized CRN versus the PU protection level is illustrated in Fig. 7(a) for $\bar{\gamma} = -15\text{dB}$. Here, a 0.90 protection level is equivalent to a global probability of detection of 90%. The sample budget is set at $\tau = 10000$, and the sensing window is set such that:

- Each SU uses $N = 2500$ samples in the basic CRN



(a)



(b)

Fig. 6. Network-based reconfiguration: (a) In the centralized approach, the FC allocates the channels to the SUs. For instance, in the basic CRN, the FC allocates f_1 and f_2 to SU_1 and SU_6 , as shown. In the network-based reconfiguration, the FC will allocate the best channels at each SU's location; (b) In the distributed approach, SUs will sense the best channels. For instance, it is shown here that f_1 and f_5 are the best channels at SU_9 . Then, SU_9 will exchange information with SUs that sense these channels. That is, SU_9 will exchange information about f_1 with SU_1 (in a single-hop) and SU_3 (in multiple hops), and so on.

scenario.

- Each SU uses $0 \leq N_m^{k*} \leq 2500$ samples in the device-based and centralized network-based reconfiguration scenarios, where N_m^{k*} is optimized together with the thresholds to maximize the throughput of SU_k at the m -th channel.

It is clear from Fig. 7(a) that the higher the reconfiguration level in the network, the higher the aggregate throughput. For instance, SUs that use the MJD achieve a lower throughput than SUs that use the RMD. This is because the former uses a single-type sensor (i.e., only the parameters of each sensor are reconfigured), whereas the latter uses sensors of multiple types (i.e., the sensor type and its specific parameters are reconfigured). More throughput gains can be accomplished when the FC reconfigures the sensing tasks for the SUs. Furthermore, for the same level of network reconfiguration, it is observed that there is a significant gain when SUs sense two channels (i.e., $I = 2$) compared to the single-channel case (i.e., $I = 1$). For instance, when the PU protection level is 90%, the throughput can be improved by 67% when multiband sensing is used in the network-based reconfiguration compared to the single-channel case. This illustrates the superiority of multiband spectrum sensing over single-band techniques.

Fig. 7(b) shows the aggregate throughput of the network-based reconfiguration as a function of $\bar{\gamma}$. The false alarm requirement for each channel is 10%. In all scenarios, a fixed sensing window is implemented, where $N = 5000$ samples. In the network-based reconfiguration, it is assumed that the SUs use the RMD, where only the thresholds are optimized. It is clear that the distributed reconfiguration approach outperforms the centralized reconfiguration. This is expected since the latter implements the OR-logic rule.

Fig. 8 illustrates how SUs reach consensus for different channels even though the initial measurements vary between SUs due to the random condition of the wireless environment (a snapshot of the network is shown in Fig. 6(b)). Because f_2 is being sensed by SUs using PDs, they will exchange information about the correlation level (shown in the top plot), whereas SUs sensing f_3 will exchange information about the energy level because this channel is being sensed using EDs (shown in the bottom plot). In both cases, it can be observed that SUs quickly reach a consensus.

6) *Complexity*: Distributed network-based reconfiguration is more complex than the other approaches, yet no centralized units are required. Because learning is done in a distributed way, the burden is shared among the nodes, and hence the additional complexity is worth it given the tremendous saving in terms of throughput and energy consumption.

In what we have discussed so far, network-based reconfiguration has been limited to the PHY layer. The following section covers reconfigurable networks where all layers of the protocol stack are jointly reconfigurable.

III. RECONFIGURABLE NETWORKING

Networking functionalities need to support a variety of issues such as mobility, QoS, security, energy-efficiency, etc. Several research directions have tried to achieve these goals.

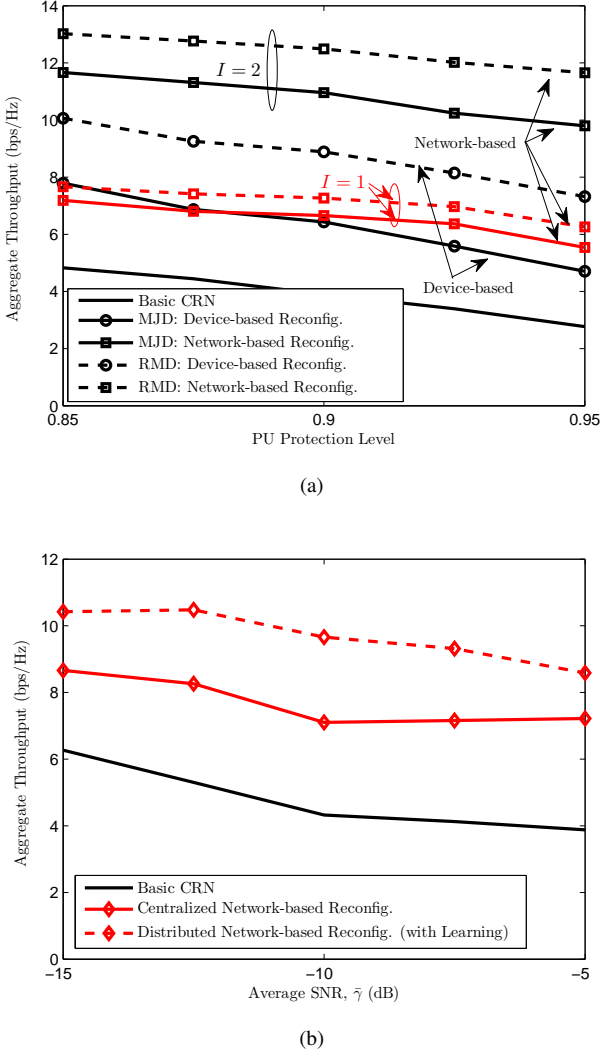


Fig. 7. (a) The aggregate throughput of the centralized CRN with different levels of reconfiguration versus the PU protection level; (b) Comparison of the aggregate throughput between the centralized and distributed network-based reconfiguration versus the average SNR.

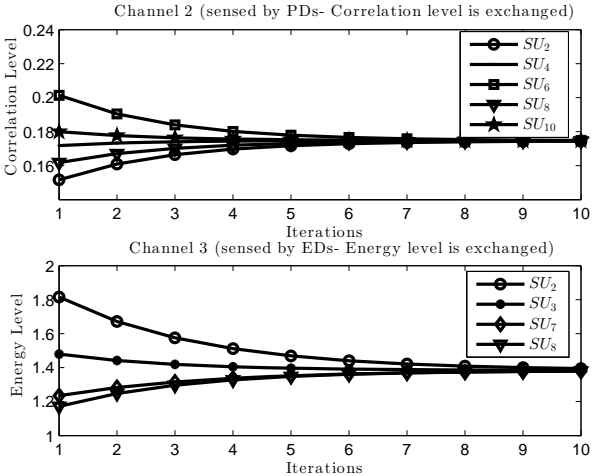


Fig. 8. Learning curves for different iterations ($\mu = 0.1$).

In this section, we focus on three areas of reconfigurable networking. These are: Cross-layer-based reconfiguration, network management, and SDNs.

A. Cross-layer-based Reconfiguration

Cross-layer design reconfigurations break the boundaries between layers by exchanging parameters or jointly designing layers in order to overcome the limitations and the rigidity of the traditional protocol stack (or single layer design) [4]. This opens significant possibilities for reconfiguration in networks, primarily because it allows for the consideration of information and objectives from multiple layers. For example, packet loss may be attributed to many factors such as link breakages, which can be detected at the PHY and MAC layers, or congestion, which can be detected at the network and transport layers. Even though the resulting event in all these cases is the same (packet loss), the reasons behind each case is quite different, and so the way to deal with the event must also be different.

Thus, in order to design a reconfigurable system which is capable of producing the appropriate decisions in response to some event, information from multiple layers may be needed. Cross-layer protocols can be classified into two main categories according to how information is exchanged between layers and how the network is organized [43].

In the first category, cross-layer protocols exchange information in one of two ways: Either by utilizing a database where parameters from all layers are stored and can be accessed by any layer, or by allowing direct exchange of information between any two layers. Having a database of parameters simplifies information exchange and reduces the chance of conflict when two layers are modifying the same parameter. However, the design of such a multi-layer database may require modifications at multiple layers. On the other hand, allowing direct information exchange allows for more localized cross-layer implementations, but may produce conflicts when the number of layers and parameters involved increases.

In the second category, cross-layered protocols are designed according to the type of network architecture. Here, cross-layered approaches can be implemented using centralized or distributed network architectures. In the centralized approach, network-wide information is used, which may result in more optimized reconfigurations. However, this may come at the cost of extra overhead and latency associated with the required information gathering. On the other hand, distributed approaches are faster and thus more adaptive. However, the decisions produced can only be optimum in localized parts of the network.

We can classify reconfigurations within each layer into processes that are controlled or information that is passed between layers. At the PHY layer, two important processes to be controlled are transmit power and data rate. In addition, MIMO techniques have introduced other important processes such as beamforming and precoding [7], whereas CRNs require spectrum control at the PHY layer. On the other hand, information that can be passed from the PHY layer to other layers include BER, CSI, SNR, interference temperature, mobility information, remaining energy, etc.

At the MAC layer, processes to be controlled include scheduling, source rate control, hybrid automatic repeat request (HARQ), maximum number of retransmissions, and contention window size, etc. Information that can be passed from the MAC layer to other layers include packet loss rate, queue length, packet types, frame length, etc.

At the network layer, the main process to be controlled is routing, which includes path discovery and maintenance. Other processes include admission control, which is particularly challenging in distributed networks due to the lack of complete information about interfering nodes. Congestion control and QoS support also coincide at the network layer. Parameters that can be passed from the network layer include path length and cost, session blocking and dropping rates, and path quality (in terms of any parameters under consideration). There is also some consideration of the transport layer, where processes include transmission window, flow control, and information passed includes error rate and congestion detection. This list of processes and parameters is not exclusive, but intended to provide an idea about what can be controlled at each layer.

To illustrate, several cross-layer implementations that enable reconfiguration in networks can be found in long-term evolution (LTE) and LTE-Advanced (LTE-A) technologies [44], [45]. One example is coordinated multipoint transmission and reception (CoMP) [46], expected to appear in Rel. 11 of LTE. This technology tries to improve performance at the cell edges by taking advantage of the fact that users are receiving signals from more than one eNodeB. There are mainly two types of CoMP: Coordinated scheduling/beamforming (CS/CB) and joint processing (JP) [47]. In CS/CB, the user is still associated with a single eNodeB, but the neighboring eNodeBs exchange information in order to coordinate their activities and reduce inter-cell interference. In CS, the exchanged information is used to coordinate scheduling to different users at the cell edges, while CB uses the information exchange to adjust the antenna beams to users. On the other hand, in JP, the neighboring eNodeBs exchange information in order to coordinate simultaneous transmissions to the same user.

The reconfigurations done in different CoMP types are shown in Fig. 9. These techniques represent a clear example of cross-layer reconfiguration of network parameters and systems in order to reduce interference and improve performance [8]. This is done based on feedback from the user equipment (UE) that indicates CSI to the eNodeBs. This CSI can be used to assign resource blocks (RBs - the smallest individual scheduling unit that can be assigned to users and consists of a chunk of adjacent frequencies) in CS, reconfigure precoding matrices for beamforming in CB, or perform JP. In any case, multiple layers are involved in the process and the system behaves differently according to the number of UEs and their CSI.

LTE has other examples where cross-layer reconfiguration improves user experience such as downlink and uplink packet scheduling [48], which relies on feedback for the UE about CSI to perform time-domain and frequency-domain scheduling (deciding the set of users that will transmit in a particular time slot and the RBs that will be assigned to each one of those users). This is also an example of joint design between the

PHY and MAC layers. Note that this is possible only for UEs with low mobility. In high mobility, CSI is changing quite rapidly and the eNodeBs will not be able to change the assignment of RBs for all users that quickly. Thus, the protocol is reconfigured according to the statuses of the UEs: If mobility is low, then frequency-domain scheduling is applied in order to assign the optimum RBs to each UE according to their CSI; otherwise, RBs are assigned over a broader part of the bandwidth to utilize frequency diversity in case of high mobility.

The above examples represent centralized implementations of cross-layer reconfiguration without the existence of a database of parameters. Centralized implementations are straightforward in cellular networks due to the presence of base stations (BSs) that perform most of the processing on behalf of all UEs in the network. Thus, fine-grained optimizations, such as assigning RBs to users based on their CSI, can be performed. For example, [49] proposes an uplink scheduler for LTE-A networks that exploits multi-user-MIMO (MU-MIMO) techniques. For cross-layer reconfiguration between the PHY and MAC layers, MU-MIMO reconfigures matrix precoding techniques to allow multiple users to use the same RBs without interfering with each other. Here, the eNodeB receives feedback from every UE about CSI for each individual RB. Based on this feedback, an optimization problem is formed where the objective function is to maximize the rate assigned to every user for every RB. This is subject to multiple constraints such as decodability (the rates have to be decodable by the eNodeB), one precoder matrix per user, finite buffers per user, limits on inter-cell interference, and a maximum of two chunks of contiguous RBs for every user in any time slot. The output of the optimization function is the set of

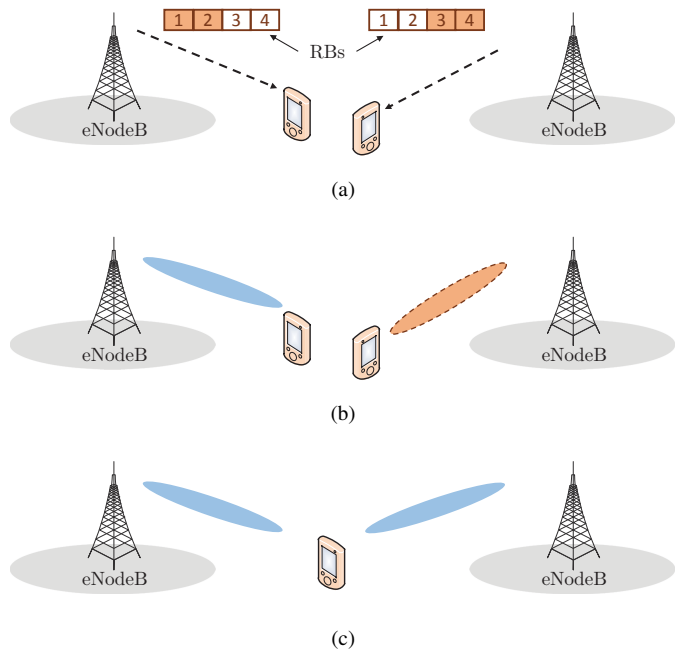


Fig. 9. CoMP techniques: (a) In CS, the scheduling is coordinated; (b) In CB, antenna beams are coordinated; (c) In JP, the user simultaneously receives information from the neighboring eNodeBs.

RBs and the precoding matrix that will be assigned to each user. Therefore, the assigned RBs and the antenna precoding matrices are reconfigured according to the CSI of each user.

The performance of MU-MIMO and CoMP techniques is analyzed in [46]. Two particular cases are considered: Homogeneous and heterogeneous networks. In homogeneous networks, all BSs belong to the same power class. On the other hand, heterogeneous networks use low-power nodes known as pico cells. The percentage gain in throughput is evaluated in case of MU-MIMO, CS/CB, or joint transmission (JT), which is the most common JP scheme. All nodes are assumed to have a full buffer and thus can continuously transmit data. The case of the homogenous network is illustrated in Fig. 10. As the figure shows, all techniques achieve a minimum of 100% throughput gain. JT achieves the highest gain, particularly at the cell edges. The gain is higher in heterogeneous networks, since the introduction of pico cells means that there is a larger number of cells and thus a larger number of cell edges [46].

Additional examples include the work in [50], which proposes a scheduler for multicast services in LTE systems. This work exploits multi-user diversity and uses AMC to maximize throughput. Another example is the work in [51], which is a cross-layer channel selection scheme for WiMAX networks. It addresses the performance degradation associated with high mobility and proposes a channel selection scheme that utilizes AMC. A reward-based scheme is used for power control in order to reduce interference. Prioritization for users is considered as well.

Centralized cross-layer design has also been used in networks without infrastructure such as ad hoc networks. However, one or more nodes have to assume a central or manager role. Thus, in order for this central node to perform the required processing, information typically has to be collected from the nodes under its service. This is not as straightforward as in cellular networks, where dedicated channels are used to relay the required information. In addition, nodes in cellular networks are synchronized, and all communications are single-hop (from UE to BS or BS to UE). Some of the challenges in ad hoc networks include: First, the lack of synchronization could mean collisions when relaying information, especially

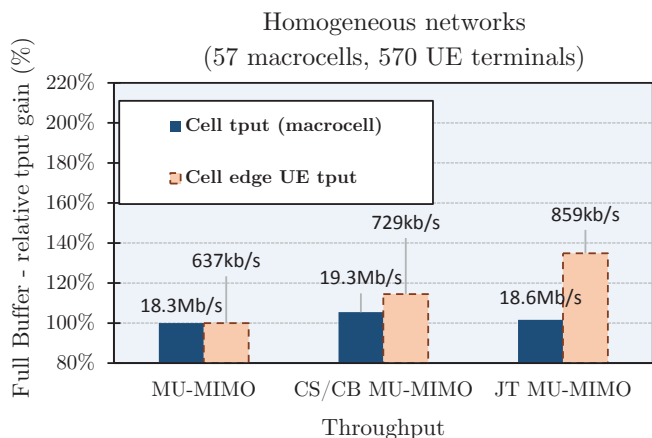


Fig. 10. Performance analysis of MU-MIMO and CoMP techniques [46].

when the number of nodes served by the central manager increases. Second, multi-hopping complicates several issues such as routing, admission control, multiple access, and third the lack of dedicated channels mean that control information has to compete for bandwidth with data. This means that the protocols may not be highly reactive and latency may be incurred in reconfiguring system parameters.

For these reasons, centralized cross-layer implementations in ad hoc networks often target long-term planning or processes that do not require quick or frequent reconfigurations. An example is the cross-layer protocol proposed for ad hoc networks in [52], which targets the optimization of throughput in sensor networks, where the existence of one or more BSs simplifies the implementation of centralized protocols. The protocol in [52] is divided into two parts: The first one performs topology control while the second one performs joint routing and rate control to maximize throughput. Two algorithms are proposed for topology control based on linear programming problems that utilize power control. In the first algorithm, the linear programming problem is solved to minimize total transmission powers at all nodes while guaranteeing connectivity. The second algorithm configures

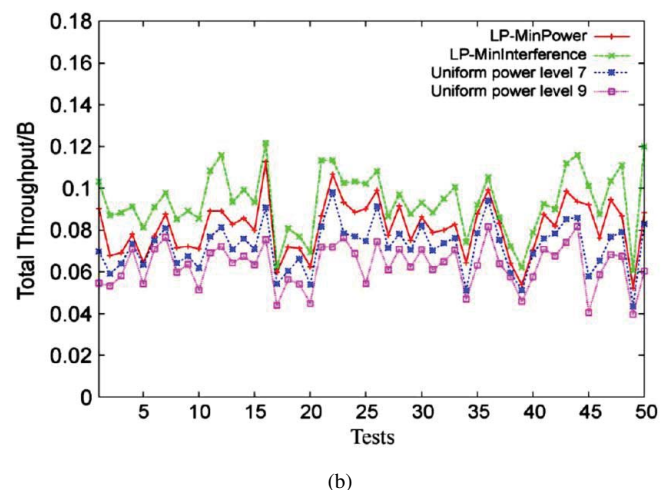
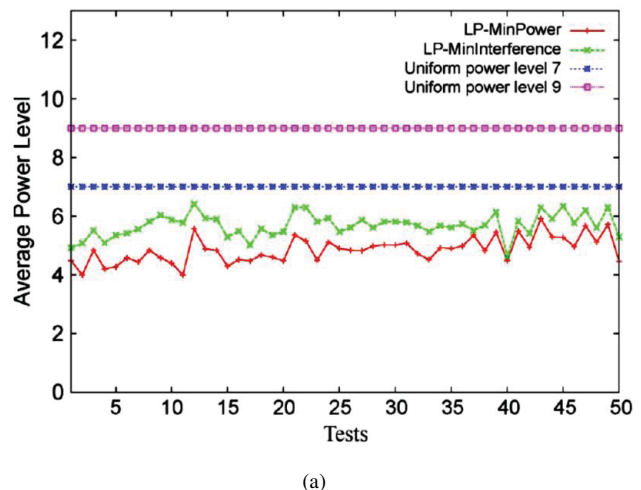


Fig. 11. Performance of the power control algorithms in metrics of (a) average total power and (b) total throughput [52].

transmit powers to minimize the total interference in the network. After the topology has been configured, the second part of the algorithm runs and another linear programming problem is solved to maximize data rates over all links. The algorithms are compared to the case where there is no power control. The results in Fig. 11 show that these algorithms have better energy efficiency and throughput.

As we can see, the protocol can only be implemented in a centralized node, where network-wide information is available. In addition, each time the algorithms are used, they are reconfigured according to the status of the nodes in the network. However, the work in [52] does not consider the latency involved in gathering this information and does not indicate how quickly the system may react to network changes.

From the above discussion, it is clear that centralized cross-layer implementations may not be suitable for highly dynamic networks, where protocols are required to be highly adaptive. For example, in vehicular ad hoc networks (VANETs), high vehicle mobility complicates the design of networking protocols. For example, routing protocols have to recover quickly due to frequent link breakages (links may not be alive for more than 30 seconds), which makes heuristic approaches such as greedy routing [53] more favorable than optimized solutions, even if greedy routing produces suboptimal results. One protocol that employs greedy routing is the grid-based predictive geographical routing (GPGR) protocol [54]. This protocol uses global positioning system (GPS) devices on board vehicles to route packets between road segments. In each segment, the protocol tries to forward data to the farthest node within its communication range. GPGR uses information about position, velocity, and direction of nodes to choose the next relay node along the path in a dynamic way that reduces link breakages and packet loss. A similar approach is adopted by reliability-improving position-based routing (RIPR) [55]. Optimization is used to study cooperative routing, but no analysis of the adaptability of the protocol is given. In all these routing protocols, the devices have to reconfigure their routing paths faster than the speed at which the network topology is changing.

Reconfiguration is particularly important in heterogeneous networks where topologies, architectures, devices, and goals are different. In the following section, we detail the case of cross-layer reconfiguration in M2M networks, which are expected to be an important part of next-generation networks.

B. Case Study: Cross-layer Reconfiguration in M2M Networks Based on Game Theory

M2M networks have many applications which can lead to new business models and opportunities. These applications include smart grid, intelligent transportation systems, healthcare, smart houses, and environmental monitoring. The deployment of indoor/outdoor M2M devices depends on the specific application. Recent studies show that about 80% of M2M communication take place indoor and 20% outdoor [56], [57]. M2M communications face many challenges such as the massive deployment of devices, transmissions in dynamic environments and architectures, high data traffic, and strict

QoS requirements (such as coverage, latency, bandwidth, lifetime, and scalability) [58]. Moreover, machines have to operate autonomously with minimal human intervention. Therefore, system reconfiguration and self-organization are required in order to meet user requirements in a cost effective way [59]–[61]. This section presents a cross-layer framework for M2M device reconfiguration based on a game-theoretic approach.

1) *Network architecture*: This section adopts the wireless M2M architecture presented in Fig. 12, where all devices are assumed to use cognitive radio (CR) technology. For indoor communication, many technologies can be used for short range communications such as WiFi, ultra wide band, mm-Wave communications, radio frequency identification (RFID), femtocell, Bluetooth, ZigBee, visible light communication, etc. [56], [57], [62]. For outdoor communication, indoor BSs for each access network can directly or indirectly communicate with antennas installed outside, on the top of buildings, or in streets in the case of residential areas. These antennas will then communicate with the outdoor macro cell BSs (eNodeBs) [63].

2) *System model*: Let N_P be the number of cognitive M2M devices and N_A the number of access networks available for uplink transmission in M2M communications. Each access network b has α_b channels in each cell, where the total number of cells in each access network is γ_b . The system capacity for the combination of all access networks can be written as

$$\mathbb{C}^S = \sum_{b=1}^{N_A} \sum_{a=1}^{\gamma_b} \sum_{c=1}^{\alpha_b} B_c \log_2(1 + \zeta_{a,c}^b), \quad (9)$$

where B_c is the bandwidth of channel c and $\zeta_{a,c}^b$ is the signal-to-interference-plus-noise ratio (SINR) at the M2M device on channel c belonging to cell a .

The total energy efficiency is defined as

$$\mathbb{E}^S = \sum_{b=1}^{N_A} \sum_{a=1}^{\gamma_b} \sum_{c=1}^{\alpha_b} \frac{B_c \log_2(1 + \zeta_{a,c}^b)}{P_A(d_{a,c})^\nu + P_{RF}}, \quad (10)$$

where P_{RF} is the energy consumed by the radio frequency circuit, P_A is the amplifier energy required at the transmitter,

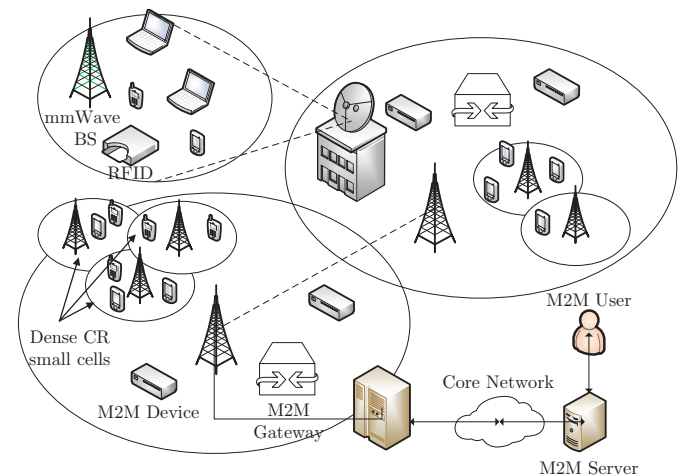


Fig. 12. Network architecture for support of M2M communication.

$d_{a,c}$ is the distance between the M2M device using channel c and BS $a \in \gamma_b$, and ν is the path loss exponent.

M2M applications have diverse QoS requirements including data rate, latency, reliability, etc. Moreover, due to the massive deployment of M2M devices, the environment and economic impact must be considered in the system design. This is because an increase in service price will increase the overall cost of the M2M communication system, and an increase in energy consumption is considered as a major threat to the environment as it is directly linked to GHG emissions.

Let SC be the set of service classes in the M2M communication system. Each class $sc \in SC$ has the following QoS parameters: Data rate (R), latency (L), reliability (RE), economic impact (EC), and environment impact (EN). Without loss of generality and for illustration purposes, we consider a limited set of service classes with QoS parameter attributes are as indicated in Table II. The parameters are defined over the levels 0 to 3 according to the significance, with 0 as unimportant, 1 as low, 2 as important and 3 as most important.

3) *Reconfiguration based on game theory*: Here a cross-layer reconfigurable (CLR) scheme is adopted for M2M device reconfiguration and resource allocation. The scheme is based on potential games where the M2M devices act as players.

A potential game is a type of strategic non-cooperative game in which all players autonomously learn their strategies [64]. There are three fundamental components of non-cooperative games which include players, strategies, and utilities. A game can be represented as $G = \{N_P, \{S_n\}_{n \in N_P}, \{U_n(\cdot)\}_{n \in N_P}\}$, where S_n is the set of strategies of the n -th player, $S = \prod_{n \in N_P} S_n$, and U_n is the utility of the n -th player, $U_n : S \rightarrow \mathbb{R}$ which maps strategy profiles S into a real value \mathbb{R} . The utility function U_n shows the satisfaction of the n -th player while considering its own strategy S_n and the strategies of other players denoted by $S_{-n} = \{S_1, S_2, \dots, S_{n-1}, S_{n+1}, \dots, S_{N_P}\}$.

In the CLR scheme, each player (or M2M device) is capable of reconfiguring its operating parameters according to the best available access network (Table III). The PHY layer is responsible for the CR module, operating frequency, and transmit power. The MAC layer is responsible for access network selection and resource allocation according to the requirement of service class mentioned in Table II. It is assumed that the network layer uses spectrum-aware routing protocols that can be classified into four classes: Power-based, delay-based, throughput-based, and reliability-based where the

routing protocol targets the energy efficiency, end-to-end delay, achievable throughput, and link stability, respectively [65] [66].

Reconfiguration is performed according to the diverse QoS requirements for classes RtRe, NRtRe, RtNRe, and NRTNRe (Table II) based on weighted instantaneous data rate, latency, reliability, economic impact, environment impact, and cross-co-tier interference. The utility function of the resource allocation and reconfiguration problem for each M2M device can be formulated as:

$$\begin{aligned} U_n(s_n, s_{-n}) = & \omega_n^R R_n^{b,a} + \omega_n^L L_n^{b,a} + \omega_n^{RE} RE_n^{b,a} \\ & + \omega_n^{RP} h_{n,c}^{b,a} P_n^{b,a} - \omega_n^{EC} EC_n^{b,a} B_n^{b,a} \\ & - \omega_n^{EN} EN_n^{b,a} P_n^{b,a} - \sum_{m=1, m \neq n}^{N_P} h_{m,c}^{b,a} \sigma_{c_n c_m} \\ & - \sum_{m=1, m \neq n}^{N_P} h_{n,c}^{b,a} \sigma_{c_m c_n}, \end{aligned} \quad (11)$$

and the potential function used for the potential game is

$$\Phi(S) = \frac{1}{2} \sum_{n=1}^{N_P} U_n, \quad (12)$$

where $R_n^{b,a}$ is the data rate, $L_n^{b,a}$ is the latency, $RE_n^{b,a}$ is the reliability, $EC_n^{b,a}$ is economic price for bandwidth $B_n^{b,a}$ corresponding to particular application class, and $EN_n^{b,a}$ is the environment cost of power $P_n^{b,a}$ corresponding to a particular application class of the n -th player in cell a of access network b , respectively. ω_n^R , ω_n^L , ω_n^{RE} , ω_n^{EC} , and ω_n^{EN} are the coefficients which intensify the QoS requirements according to the service classes mentioned in Table II. ω_n^{RP} , the reward/penalty coefficient is set to 1 when the access network is cost effective and have less impact on environment (e.g., small cells and WLAN) and is set to -1 otherwise. $h_{m,c}^{b,a}$ is the channel gain between the m -th M2M device and BS a which serves the n -th M2M device. $\sigma_{c_n c_m}$ is the interference function which is one if channel c is same for the n -th and the m -th M2M device and zero otherwise.

In this game theoretical framework, each player executes the three-step cycle shown in Fig. 13:

- In the *sensing step*, each player interacts with the radio environment for spectrum sensing to determine available

TABLE II
QoS CLASSES FOR M2M APPLICATIONS

Service	R	L	RE	EC	EN
Real-time/Reliable (RtRe)	2	3	3	0	1
Non-real-time/reliable (NRtRe)	1	0	3	3	3
Real-time/non-reliable (RtNRe)	2	3	1	2	3
Non-real-time/non-reliable (NRTNRe)	0	0	1	3	3

TABLE III
RECONFIGURABLE M2M DEVICE

Layers	Reconfigurable Parameters/Protocols
Application	QoS requirements
Network	Spectrum aware routing protocol
Data Link & Mac	Network selection Channel allocation
PHY	CR module Operating frequency Transmit power

channels and acquire their link gains.

- During the *analysis and reasoning* step, each player maximizes its utility by choosing appropriate strategies while considering the current strategies of other players. Nash Equilibrium is considered as a steady state point after which the players cannot change their strategies to maximize their utilities. It has been demonstrated that for the potential function in (12), the players reach Nash Equilibrium after a certain number of iterations. For an in-depth study of the Nash Equilibrium and the convergence of this game, the reader is referred to [64], [67].
- In the *decision* step, each device decides to update its strategy according to the best response update strategy. If the device updates its strategy then it will share this strategy with its neighboring devices.

4) *Performance analysis*: The performance of the system is evaluated through extensive computer simulations. Four service classes are considered with 25% of total M2M devices in each service class. For illustration purposes, it is assumed that cellular, WLAN, and mmWave channels are available for data transmission where the parameters for cellular bands are according to 3GPP specifications [68], WLAN are according to WiFi [69], and mmWaves are according to specifications in [69], [70]. It is assumed also that four routing protocols are available for M2M devices and these are power-based [71], delay-based [72], throughput-based [73], and reliability-based [74]. The reconfigurable approach is compared to a user-optimized scheme called joint resource allocation and network selection (JRANS), where the resources are allocated based on the interference factor and the channel gain only [75]. The performance metrics considered in the simulations are throughput, energy efficiency, economic cost, and environment impact.

Fig. 14 illustrates the convergence in terms of throughput of the CLR scheme and comparison is carried out with JRANS scheme. In the CLR scheme, each M2M device selects suitable strategy to improve its utility while considering strategies of other players. After some iterations, a steady state (Nash Equilibrium) is reached from where no one can increase its

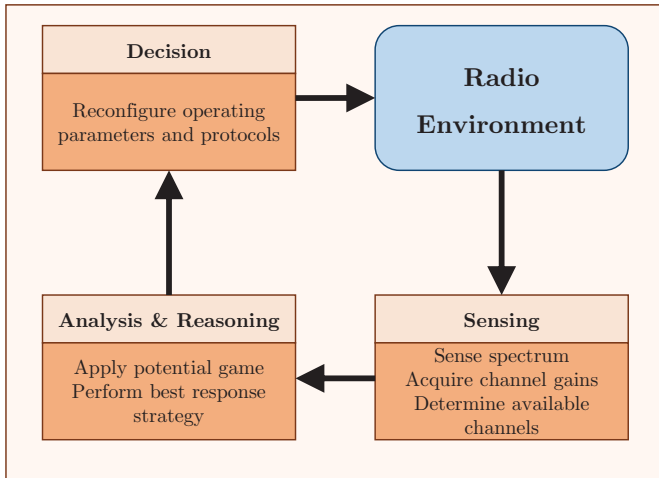


Fig. 13. Main steps in the CLR scheme.

utility. The JRANS scheme converges faster because it only considers interference and channel gains for channel allocation and network selection. However, the CLR outperforms the JRANS in terms of total throughput.

Fig. 15(a) compares the system throughput versus the number of M2M devices for the CLR scheme, JRANS scheme, and two other schemes that consider different variants of (11) (without economic and environment impact, and without reconfigurable routing). It should be noted that the higher throughput in the case of non-reconfigurable routing is due to the fact that in that scheme, throughput-based routing protocols are used for all M2M devices. However, the CLR based scheme reconfigures the routing protocols according to the corresponding service classes. The CLR scheme outperforms the JRANS scheme and the other scheme that excludes economic and environment impact from the utility function. The economic and environment impact pushes M2M devices to unlicensed and low power resources such as mmWaves, which offer high bandwidth in indoor areas while reducing energy consumption and satisfying other QoS requirements. This explains the degradation of system throughput when economic and environment impacts are excluded from utility function.

Fig. 15(b) shows energy efficiency comparison for the aforementioned schemes. The CLR scheme outperforms all other competitors. The performance degrades in terms of energy efficiency when the economic and environment impacts are removed from the utility function because in that case the system will choose channels without considering their energy footprint. Moreover, energy efficiency without considering reconfigurable routing further degrades the performance because in that case the system uses the routing protocol with the highest throughput regardless of its energy cost.

Fig. 16 compares economic cost and environment impact of the CLR and JRANS schemes versus the number of M2M devices, respectively. It can be seen that the CLR scheme has about 25% less economic cost and 27% less environment cost in comparison with the JRANS scheme when number of M2M

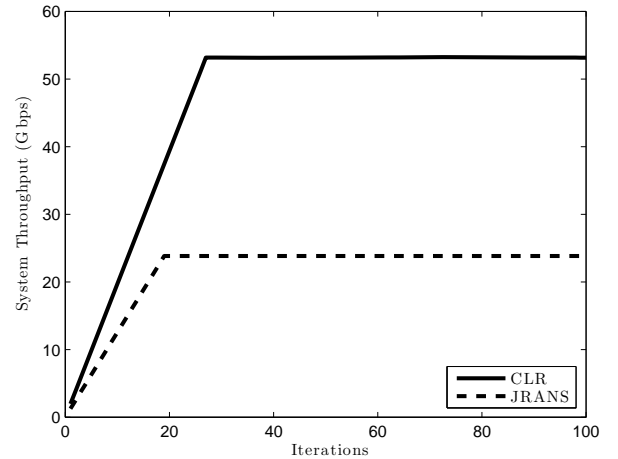


Fig. 14. Convergence of the CLR and the JRANS schemes in terms of system throughput (the number of M2M devices is 300).

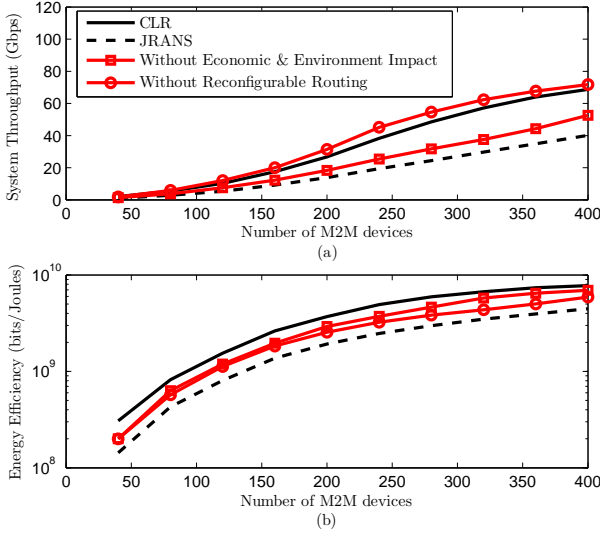


Fig. 15. (a) Throughput and (b) energy efficiency with variations of the number of M2M devices.

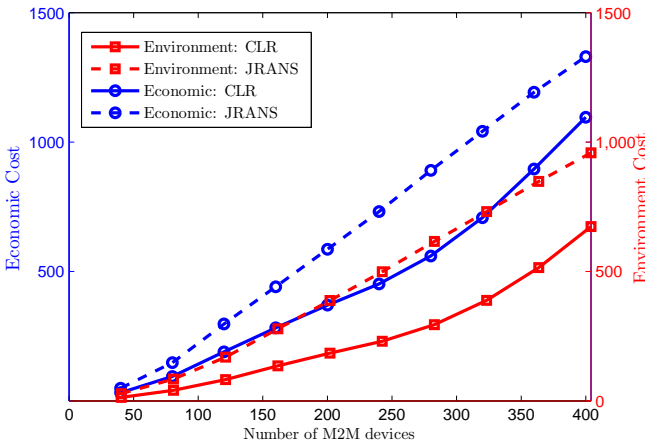


Fig. 16. Economic cost and environment impact versus the number of M2M devices.

devices is 400.

5) *Complexity*: It should be noted that the computational complexity of CLR is higher than JRANS. However, CLR uses a decentralized scheme where the complexity cost is distributed among all M2M devices. For the communication overhead (due to information exchange required by the game-theoretical approach), it can be significantly reduced if M2M devices use existing common control channels. This complexity cost is fully justified given the significant performance improvement that CLR offers in terms of throughput, energy efficiency as well as economic and environment costs compared to JRANS.

C. Network Management

Network management covers a broad range of functions that generally have to do with network reconfiguration. These functions can generally be classified into three categories:

Parameter configuration, connectivity management, and network monitoring. Parameter configuration consists of choosing the appropriate network parameters to achieve a given set of objectives. For example, topology control algorithms typically focus on choosing the appropriate transmit power that achieves network-wide connectivity [76]. This can be done in a centralized way [76] or in a distributed way [77]. It can also be homogeneous (all nodes transmit using the same transmit power) or heterogeneous. On the other hand, connectivity management encompasses a wide variety of tasks such as mobility management [78] [79], radio resource management (RRM) [48] [80], and QoS support [48]. Finally, network monitoring refers to monitoring the links between users and the serving stations or between users. It is a function used to collect information that serves other networking functions such as CSI or mobility information.

Clearly, network management is cross-layer in nature. However, the main difference between network management protocols and other cross-layer protocols is that network management focuses on functions that have network-wide effect, while other cross-layer protocols may optimize individual links or a small set of functions. In this sense, network management acts as the brain behind the network. In this section, we will consider two examples that illustrate how network management is used across different layers. Particularly, we consider the cases of RRM in LTE, and network management in WLANs.

In LTE, RRM is performed using several protocols [44], [48]. At the topmost layer of the LTE protocol stack, the network layer, the radio resource control (RRC) protocol performs three main functions: QoS management, admission control, and semi-persistent scheduling. QoS management builds a profile for the requirements of the upcoming communication session, admission control decides if these requirements can be satisfied given the current load, and semi-persistent scheduling is used for periodic allocation of resources (used for deterministic communication sessions such as video streaming). Then, at the MAC layer, the radio link control (RLC) and MAC protocols perform three other functions: HARQ, dynamic scheduling, and link adaptation. HARQ is an efficient error control mechanism that adjusts the number of error control bits according to link quality. On the other hand, dynamic scheduling and link adaptation are performed jointly according to the algorithm presented in Section III.A. Finally, at the PHY layer we find mainly the channel quality indicator (CQI) manager, which is basically a tool to monitor link quality. Thus, these protocols perform network management functions in the categories of parameter configuration (HARQ, link adaptation, dynamic scheduling, and semi-persistent scheduling), connectivity management (QoS management and admission control), and monitoring (CQI management). The reconfigurations performed by these protocol stack are numerous. For example, whether the system will apply semi-persistent scheduling or not depends on the type of application. In addition, admission control considers existing connections as well as new requests to determine the ones that will be dropped. Furthermore, dynamic scheduling may switch between scheduling techniques according to the type of applications requested by the users. Generally speaking,

the main objective of the reconfigurations in the LTE protocol stack is QoS support. The protocols are designed to consider a large number of users with varying application requirements.

RRM functions in LTE are performed at the eNodeBs, which control the operations of the UEs in their service areas. In addition, in LTE-A, eNodeBs communicate with each other and coordinate their transmissions, as explained before. In any case, RRM functions are coordinated and can be optimized even across cells. However, this is not the case in WLANs with multiple interfering access points (APs). Here, management functions are performed independently at each AP, with minimum collaboration with the other APs. Nevertheless, there are provisions for network management within the IEEE 802.11 family of standards. For example, 802.11r [81] allows for fast negotiations between APs by allowing multiple negotiation sets to occur in parallel, 802.11k [82] defines how APs exchange management information, while 802.11s [83] allows APs to operate in a mesh topology. However, each AP still manages the users within their cells independently of any other APs. These management functions include channel monitoring, priority management, and power configuration. However, there are two particular management functions that require collaboration between APs: Channel selection and AP association. The first is the process by which the AP chooses the channel over which it will operate, while the second is the process by which the user chooses a serving AP. Traditionally, channel selection is done by considering the channels utilized by the neighboring APs, in order to reduce inter-AP interference. Once the AP chooses the channel, it starts transmitting its beacon messages to announce its presence. Then, users typically associate with the AP with the highest signal strength. It is worth noting that other protocols for AP association have been proposed that consider metrics such as packet error rate [84] or throughput [85], which are more efficient than signal strength alone.

According to [86], channel selection and AP association must be done in a user-centric way in order to improve throughput. This means that the channel conditions seen by the user must be considered, and load balancing must be performed. Thus, the authors in [86] propose a user-centric protocol for network management in WLANs. This protocol starts in the traditional way by having the AP select a channel and send beacons. Then the user sends its CSI seen with every AP from which it is receiving beacons. The APs then calculate the potential throughput according to this CSI and send the information back to the user. Using this information, the user can associate with the AP that provides the maximum throughput. In addition, each AP periodically collects user performance statistics from its users and may trigger a channel adjustment operation if the aggregate throughput drops below a certain threshold. Since AP association is based on throughput, load balancing is implicitly included since loaded cells will provide users a potentially low throughput estimation, discouraging it from associating with it. These reconfigurations are illustrated in Fig. 17.

The performance of the user-centric algorithm is evaluated in [86] from the perspective of the throughput observed by the user and the throughput observed by the AP. Several

experiments are conducted. In one experiment, one user is tagged as the observing user and its throughput is measured for different number of users per unit area. This experiment evaluates the performance of the AP association scheme proposed in [86]. Fig. 18 shows that the proposed scheme achieves performance gains from 25% to about 90% over the self-organization management framework, and the gain is higher in lower user densities.

Network management protocols have also been proposed for ad hoc networks. However, due to the distributed nature of these networks, it is difficult to propose protocols that can have a network-wide effect. In addition, the multi-hop nature of these networks make collecting information costly. Thus, network management protocols for ad hoc networks typically focus on a narrow set of objectives such as topology control [77], admission control and scheduling [87] [88], or congestion control [89].

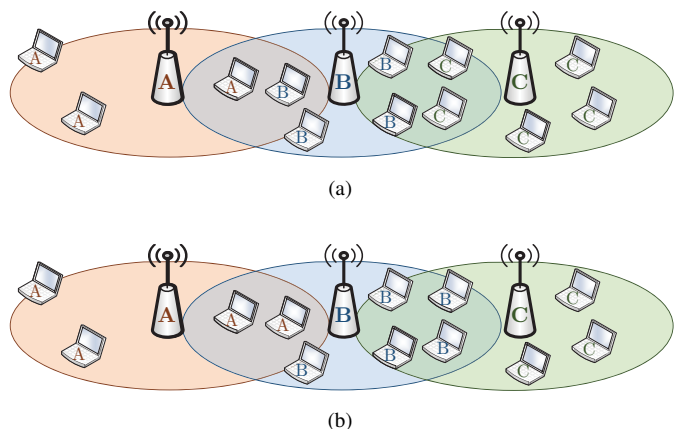


Fig. 17. Topology reconfiguration in three WLANs: (a) After initial associations; (b) After reporting their CSIs. Some device associations have been reconfigured to achieve load balancing.

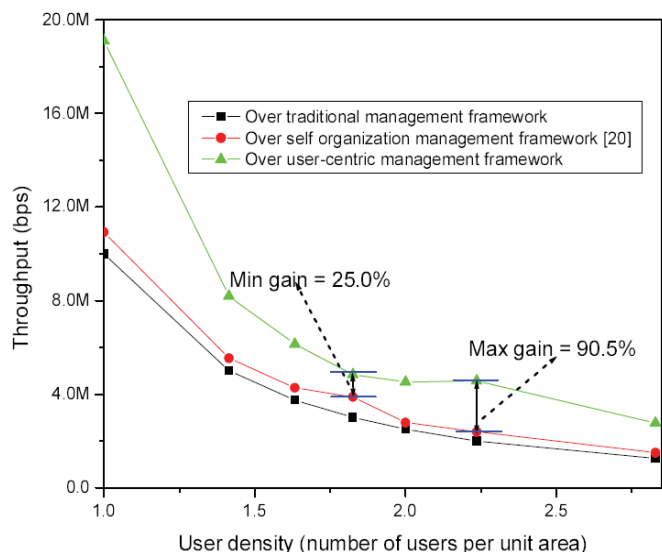


Fig. 18. Performance results for the AP association scheme.

D. Software-Defined Networking (SDN)

Conventional networks rely on protocols that implement a particular set of functions that target some specified objectives, as we have seen in previous sections. However, modern networks are becoming increasingly complex in terms of challenging requirements such as changing user demands, reliability, scalability, and security. These demands may even be time-varying. Thus, protocols (especially those that are implemented in hardware) may not adapt so easily. This has led to the revolutionary idea of SDNs. The main concept in SDN is to decouple the control plane from the data plane [90] [91]. Thus, all network policy rules and configurations, forwarding rules, packet handling rules and priorities, etc., are implemented using reprogrammable software called the *controller* [92]. These rules are forwarded to special SDN-compliant switches and routers, where they are stored in tables called *flow tables*.

SDN architecture is illustrated in Fig. 19, where there are several controller implementations such as POX [93], NOX [93], Beacon [94], and Floodlight [95]. Each one provides a base on which any application can be implemented. On the other hand, communication between the controller and the SDN-complaint devices is based on an SDN protocol such as forwarding and control element specification (ForCES) [96] or interface to the routing system (IRS) [97]. Nevertheless, the most prominent SDN protocol is OpenFlow [90], [98]. Using, OpenFlow, entries in the flow table contain match fields (forwarding rules), counters and instructions (prioritization, dropping rules, etc.). Upon receiving a packet, the SDN-compliant device checks its flow table for a match. If found, the match will be used to forward the packet. However, if a match is not found, the packet is forwarded to the controller for analysis. The controller then uses software to decide what to do with the packet, and may then forward a new matching rule to the device. Since the controller is reprogrammable, all networking policies can be re-adjusted as needed.

SDN can be implemented in many ways [99]. For example, a centralized implementation is to have one controller managing multiple devices, while a distributed implementation is to have one controller per device and to have the

controllers communicate with each other to coordinate their operations. Hybrid and hierarchical implementations (one controller managing multiple controllers, each managing one or more switches) are also possible.

Due to this flexible architecture, SDN can have many possible applications. For one thing, it allows for experimenting with protocols without having to implement on hardware. It also has huge implications on interoperability in multi-vendor networks [100] [101]. For this reason, major cloud providers (such as Google and Amazon) are highly in favor of SDNs. Using SDNs, these cloud providers can purchase equipment from any vendor, and they can operate together using instructions from the controller if they are SDN-compliant [102]. SDNs also simplify network management significantly and allow for large scale automated management. They also allow for easy management of VLANs [102], since managing network IDs and the size of each network becomes simple using the software-based controller.

SDNs also make load balancing and congestion control quite simple since the controller can get statistics from the devices and optimize network traffic distribution [103] [104]. Another important application is network security [105] [106]. This is because the presence of the controller provides powerful tools for traffic analysis. The software can also be modified and adapted to consider new forms of attacks. Thus, security attacks such as distributed denial of service (DDoS) can be handled more easily [90]. The controller also provides means for traffic isolation, simply by adjusting the forwarding and dropping rules. Finally, SDNs provide significant tools for flexible QoS support [107] [108]. Controllers can inspect packet types and payload, and adjust priorities and queues accordingly. If demand changes, the controller can re-adjust its configuration. Note that all these reconfigurations can be performed automatically and in software, a capability that cannot be easily achieved outside the scope of SDN. An illustration of some SDN applications is shown in Fig. 20.

It is worth mentioning that there are multiple versions of the OpenFlow protocol [90]. The earliest version is OpenFlow 1.0.0 and is the most widely deployed version. Here, the controller has a single flow table and there is no support for IPv6, multi-protocol label switching (MPLS), or simultaneous communication with multiple controllers. MPLS matching and multiple flow tables are included in version 1.1.0. Then, version 1.2.0 adds support for IPv6 and inter-controller communication. Finally, the latest version is 1.3.0 and has more flexible capabilities for communications between controllers.

SDN can be used in wired and wireless networks and can provide a revolutionary way for network reconfiguration and adaptability. It is certainly a very promising technology that is expected to continue rising in the near future. However, there are challenges that need to be considered, and several of them stem from the dependency of the network on the controller. This presents reliability and scalability threats. If the controller fails, the network will not function. Moreover, if the controller is managing a large number of nodes, it may only be able to handle a limited amount of packets being forwarded to it for analysis. This also makes the controller an attractive target for attackers not only because it represents a single point of

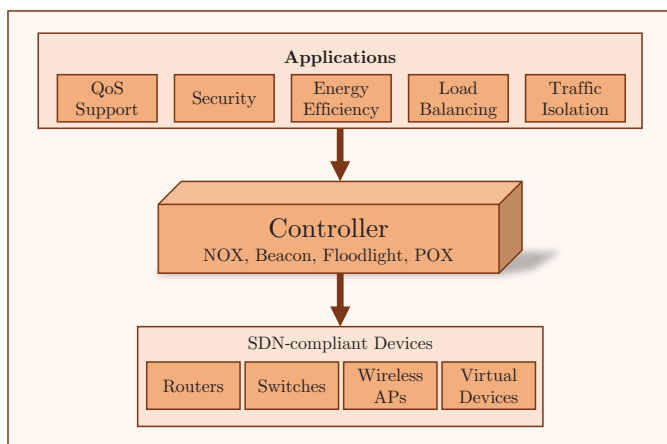


Fig. 19. SDN architecture.

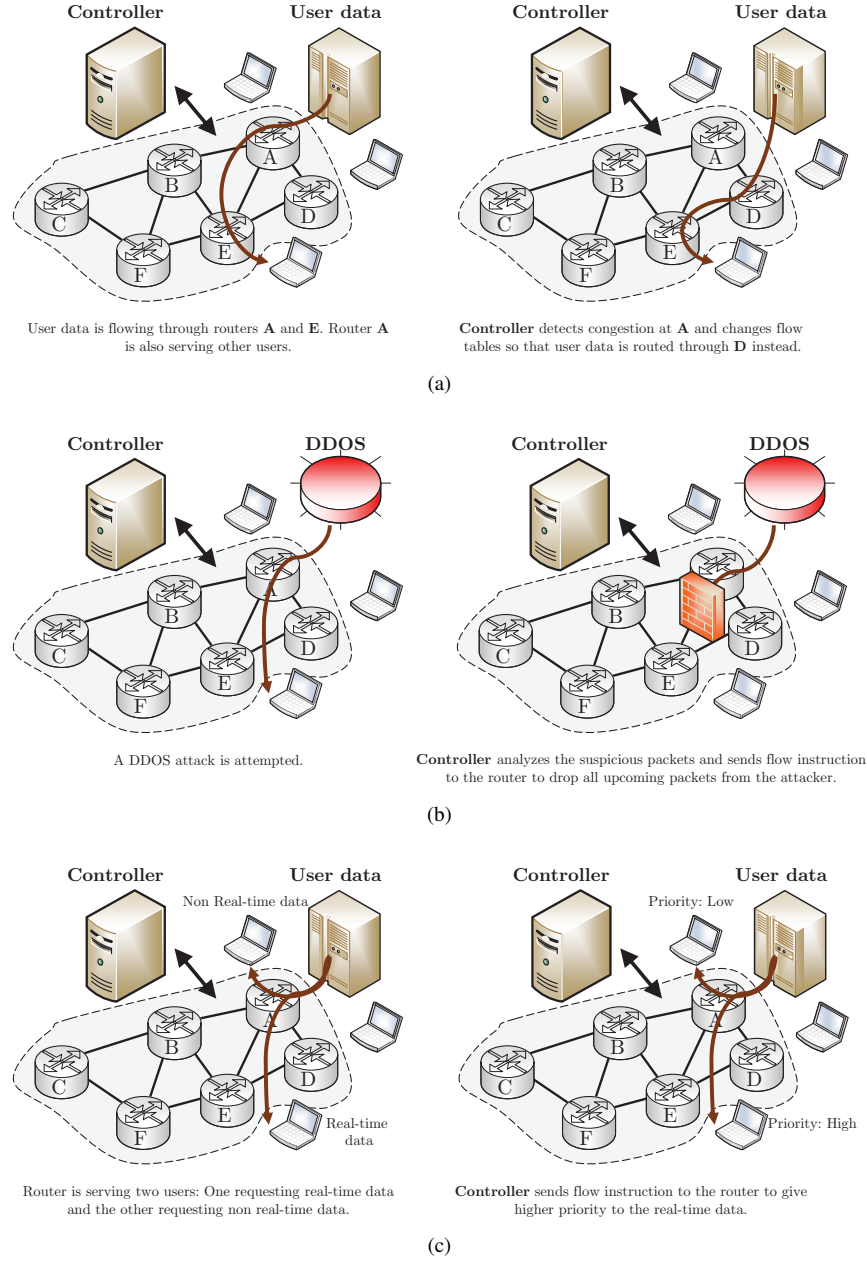


Fig. 20. Examples of SDN applications: (a) Load balancing using SDN; (b) Security using SDN; (c) QoS support using SDN.

failure, but also because it holds critical information about the network. Thus, security measures have to be in place such as intrusion detection, policy adaptation, and network forensics. Finally, recovery time of SDNs needs investigation. In order for a node to recover from an error, it needs to send the packet to the controller and receive a decision back. This may cause some latency. This has to be considered in the controller design.

IV. NETWORK INTELLIGENCE

Cross-layer reconfiguration has some limitations [109], [110]. For example, there is a risk known as adaptation loops, where improving one aspect may lead to downgrading another. There are concerns regarding the coexistence of several cross-layer designs. Since cross-layer designs are

generally application-specific, it may not be straightforward to integrate multiple solutions. Thus, wide-scale implementations are challenging.

In this section, we focus on cognitive networking and context-awareness as means of improving cross-layer adaptation and adaptability.

A. Cognitive Networking

Cognitive networking is based on the cognitive loop [111], shown in Fig. 21. It contains an observation entity responsible for monitoring the external world through sensory measurements. It also collects information about the internal state of the system. All this information is processed and passed to a reasoning machine. Here, the information is analyzed

and decisions are made. These decisions can be short-term or long-term (requires planning). Furthermore, the decisions may change the internal state of the system and may require implementation using physical actions. In addition, feedback information from different parts of the system is passed to a learning entity that builds a knowledge database. This database is used to enhance the decisions produced by the reasoning machine over time.

The cognitive loop implements degrees of intelligence in the network. Rather than targeting specific applications or algorithms, cognitive networking focuses on methods of analyzing information, whatever it may be, and producing decisions, while focusing on some specific goals or objectives. Since learning is a highly important cognitive tool in humans, it is also implemented in the cognitive networking loop. Just as humans improve their decision-making by learning from their past experiences, networks too can optimize their decisions by observing the results of their previous actions and of the changes in the external world.

The importance of cognitive networking has led standardizing bodies to develop frameworks for management and control based on the cognitive loop. Three standards in particular are worth mentioning [112]: Reconfigurable radio systems (RRS) [113], IEEE 1900.4 [114], self-organizing networks (SON) [115].

Cognitive networking has also found implementations in ad hoc networks, where the lack of centralized nodes poses challenges. Several tools of computational intelligence (CI) have been utilized for this task. For example, evolutionary algorithms such as particle swarm optimization and ant colonies have been used in [116], [117] to perform topology maintenance and congestion control in wireless sensor networks (WSNs). Particle swarm optimization considered node positions and data transfer rate, while ant colony optimization considered the time taken to traverse a path and the number of successful iterations in the search process. Simulation results show that the proposed algorithms are able to reduce congestion and delay and increase throughput. Other CI tools that have been used include game theory [118]–[120] and Bayesian Networks [121]–[123].

Learning transforms the operation of the system from policy-based to intelligence-based [124]. Policy-based systems

follow static rules that are predefined and hardcoded, while intelligence-based systems have the ability to learn new states and evolve the operation of the system to achieve optimal or near optimal performance. These systems also have the ability to detect statistical patterns in the environment, such as regions with good or bad channel conditions or the nature of the traffic transmitted and adapt to them. Since most wireless systems exhibit dynamic properties, it is clear that learning can be a powerful tool for improving system performance.

There are mainly three categories of learning algorithms [125], [126]: Supervised learning, unsupervised learning, and reinforcement learning (RL). In supervised and unsupervised learning, examples or training sequences are used to accomplish the learning process. Learning agents use these sequences to identify patterns or recurring behavior and learn about the system. In supervised learning, each entry of the training sequence consists of an input signal and a desired output value, and the job of the learning protocol is to find an inferred function in all the sequences that can be used for calculating new values. On the other hand, the desired output is unknown in unsupervised learning. The learning algorithm simply observes the input and tries to detect patterns. In RL, the training sequence is not needed. The learning agent relies on feedback from the normal interactions of the system and identifies the actions that yield the most reward or least cost.

RL is the most prominent learning tool in wireless networks. This is because RL is a simple and model-free algorithm [124]. This means that RL does not need to model the entire operating environment in order to perform learning, only the desired parameters are monitored. This also means that RL can focus on metrics that enclose a wide range of networking issues, rather than having to learn about all the parameters in the network. For example, the RL algorithm can monitor end-to-end delay in order to discover the states that minimize this metric rather than monitoring a wide range of issues such as queue length, routing metrics, channel quality, etc. Delay, not only encompasses all these issues, but also has direct impact on the QoS observed by the user.

RL has been used in various ways in wireless networks and at different layers. For example, RL is used in [127] to design an opportunistic routing protocol. The learning protocol rewards paths with good transmission success probabilities in order to minimize the cost of relaying a packet. In addition, this protocol balances the exploitation of paths that have been given previous rewards versus the exploration of new paths in order to improve the efficiency of routing decisions. This is done using probe packets that are sent periodically to explore new paths. In another example [128], RL is used to design a MAC protocol where nodes cooperate to maximize the throughput per unit of consumed energy. Nodes exchange the RL values in order to explore the efficiency of the decisions from a network-wide perspective. Thus, nodes adjust their transmission power and transmission probabilities according to the accumulated rewards. RL has also been used in [129] to improve the efficiency of spectrum sharing in CRNs. A warm-up stage is defined where nodes test the quality of different channels. Afterwards, the nodes will only learn about the set of channels that are discovered to be promising during the warm-

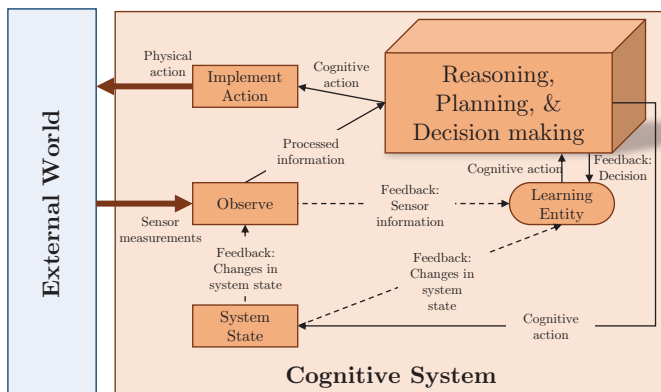


Fig. 21. Cognitive loop [111].

up stage. This reduces the sensing frequency and improves the quality of transmission by focusing on good channels. In [130], RL is used in defining energy management policies for sensor nodes with finite buffers. The goal of this scheme is to maximize network lifetime in the presence of energy harvesting sources.

It is worth mentioning that RL has challenges. First, exploration versus exploitation has to be addressed. Exploration may lead to better solutions, but it also means slower algorithm convergence. In addition, traditional RL algorithms have to be implemented either in a centralized way or at every node in the network. Some modifications have been proposed in order to enable distributed implementation. Examples are the distributed value function (DVF) [131] and the distributed reward and value function (DRV) [124], which combine the values learned at different agents. Nevertheless, these functions require the exchange of such values, thus incurring higher overhead. Finally, there is a trade-off between learning time and granularity of the learning process. Having more states may mean that accuracy of the learning process is increased. However, it also means that it may take longer for the protocol to visit all these states and learn about their performance.

There have also been limited efforts on using supervised and unsupervised learning in WSNs. For example, supervised learning is used in [132] to design a neural network that exploits the spatial correlation of sensed events in nodes with close proximity in order to reduce packet loss. However, the algorithm has to be implemented in a centralized way. On the other hand, unsupervised learning is used in [133] to classify SUs and PUs in CRNs. The work also considered malicious attacks from SUs that misclassify themselves as PUs. In another paper [134], Bayesian learning is used as a method for unsupervised learning in support to a game-theoretical spectrum sensing technique. An auctioning game is proposed where nodes entered *bids* (to a centralized coordinator) on using the available spectrum holes. Bayesian learning is used to determine whether a node should enter a bid in each round or not, based on previous experience. However, as mentioned before, supervised and unsupervised learning algorithms often do not adapt quickly enough in dynamic environments. In addition, if goals or conditions change, the training sequences may have to be redesigned.

B. Case Study: Network reconfiguration based on reasoning and learning for WSN

In WSNs, there are numerous issues that have to be managed while performing planning and negotiations [3], [110]. Connectivity, coverage, sensing frequency, QoS requirements, routing, packet loss rate, and congestion are among those issues. All tasks have to be performed while ensuring energy efficiency. Furthermore, WSNs may be highly dynamic: Sensor nodes may be mobile, duty cycling may continuously change network topology, the wireless environment may be changing, etc. Thus, managing network resources and achieving the goals become highly challenging. Since most of these issues may involve multiple layers, cross-layer approaches have been typically involved. However, despite the different approaches

that have been tried, there is still a need for a tool that can address these challenges as well as the dynamic nature of the network. The characteristics of such a new tool may include the ability to adapt to changing goals, addressing conflicting constraints, supporting QoS requirements, and being able to be implemented in a distributed way.

1) *Basic concepts:* A mathematical tool known as weighted cognitive map (WCM) has been recently introduced [135], [136]. It presents significant promises in achieving the aforementioned requirements. A WCM is a graphical model that can represent any dynamic network using the underlying causal relationships between the different parts of the system. Vertices of the map represent concepts (such as transmit power control or BER from the real system). A characterizing scalar, A_i , is associated with each vertex, C_i , which identifies its activation level. A_i can take any real value from $[-1, 1]$. Negative values imply a decreasingly active concept (for example packet loss ratio (PLR) becoming lower), positive values imply an increasingly active concept, while a value of zero imply that a concept is inactive. On the other hand, edges of the WCM connect concepts that are causally related. Weights are associated with each edge to identify the strength of the causality. These weights can also take any real value from $[-1, 1]$. Changes in the activation levels of the concepts, which can occur due to changes in the environment (such as an increase in PLR), trigger the WCM to operate. The inference properties then enable the WCM to calculate new activation levels using simple mathematical operations. The new activation levels translate into actions that are executed by the network (for example, an increase in the activation level of PLR may lead to an increase in the activation level of transmit power, which causes the system to use the next higher transmit power).

As an example, the WCM in Fig. 22 models the relationships between six processes (concepts) affecting a node, namely, transmit power, data rate, maximum number of retransmissions allowed before a packet is dropped, BER, throughput, and the expected transmission time (ETT). The edge weights shown represent the strength of causality between concepts. We can classify concepts in any WCM into end-to-end goals, environment variables, and processes. All processes interact to achieve the end-to-end goals. On the other hand, environment variables cannot be manipulated directly, but only as a result of other actions. It can be seen from Fig. 22 that the end-to-end goal is throughput (C5), since all other concepts can cause changes to it. It can also be seen that there are two environment variables considered, namely BER and ETT, since they cause changes in other processes but cannot be changed directly by them. They can only be affected by changes in the environment or as a result of actions taken by the WCM. The processes available to the WCM are transmit power, data rate adaptation, and changing the maximum number of retransmissions.

The WCM can be represented in a matrix form as

$$W = \begin{matrix} & \begin{matrix} C_1 & C_2 & C_3 & C_4 & C_5 & C_6 \end{matrix} \\ \begin{matrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0.9 & 0 \\ 0.3 & -0.5 & 0.8 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4 & 0 & 0 & 0 & -0.7 & 0 \end{pmatrix} \end{matrix}. \quad (13)$$

For a WCM with N_C concepts, its status at time t can be given by

$$A(t) = [A_1, A_2, A_3, \dots, A_{N_C}]. \quad (14)$$

The inference process of WCMs is the one by which the values of concepts change according to their underlying causal relationships. Thus, according to the inference properties of WCMs, the status at time $t+1$ can be given by

$$A(t+1) = f(A(t)W), \quad (15)$$

where $f(t)$ is a threshold function that determines the type of the WCM. The inference process is initialized when a particular concept is triggered, causing its activation value to change (e.g., a sudden increase in BER). Therefore, the input to the WCM at time t is $A(t)$, including the new value of the concept that is just triggered. The triggered concept influences other concepts according to W , thus producing an output to the WCM, $A(t+1)$, as given by (15). Thus, WCMs can be implemented in a distributed way and has powerful reasoning capabilities. In addition, since WCMs are based mainly on the causal relationships between concepts, conflicting constraints can be easily considered. Furthermore, if goals or constraints change, modifying the edge weights is sufficient to allow the WCM to adapt to the new requirements.

In this case study, we will illustrate how WCMs can be used to design a reasoning machine for WSNs [137], [138]. For the WSN architecture, a cluster hierarchy is employed for its popularity and simplicity. The WCM is implemented at the cluster heads (CHs) and a sink node, which are assumed to have higher energy and processing capabilities than regular sensor nodes (Fig. 23). Sensor nodes are deployed in uniform random positions across the area. Thus, the WCM at each CH

is responsible for determining the parameters to be used by all nodes in its cluster. It is assumed that nodes are synchronized and that time is slotted. Every T consecutive time slots, the WCM determines the appropriate number of nodes that must be activated in order to guarantee connectivity and coverage, using the theorem that is proposed in [139]. These nodes remain active for the next T time slots. On the other hand, the WCM monitors the cluster every T/η time slots to check if actions need to be taken. The value η depends on the dynamics of the network. In highly dynamic networks, η can be equal to T so that the WCM is activated in every time slot.

The first step in designing the WCM is to specify the end-to-end goal(s). Since network lifetime and throughput are the two main goals of the WSN, they must be directly incorporated as goals of the WCM. This way the WCM can constantly monitor these goals and make sure they are constantly achieved. Therefore, in order to consider network lifetime, one goal of the WCM will be energy consumption. Given that the initial battery power of nodes is E mAhr, and that a target lifetime, X time slots, is defined by the user, the task of the WCM is to make sure that the rate of energy consumption does not exceed E/X mAhr per time slot per node. However, due to the expected variation in the energy consumption of nodes, forcing the WCM to react whenever there is an increase in energy consumption in a time slot will result in frequent changes, which is undesirable. Thus, if N_C is the number of nodes within the cluster, the WCM at the CH will monitor the energy consumption, EnC_i , within its cluster and makes sure that the total energy consumption satisfies

$$\sum_{i=1}^{N_C} EnC_i < \frac{T \cdot N_C \cdot E}{4X}, \quad (16)$$

mAhr per $T/4$ time slots, where monitoring takes place every $T/4$ time slots so that the WCM has enough time to take action before the next scheduling round takes place. Similarly, if N_{WSN} is the total number of nodes in the network, the sink node is responsible for making sure that the total energy

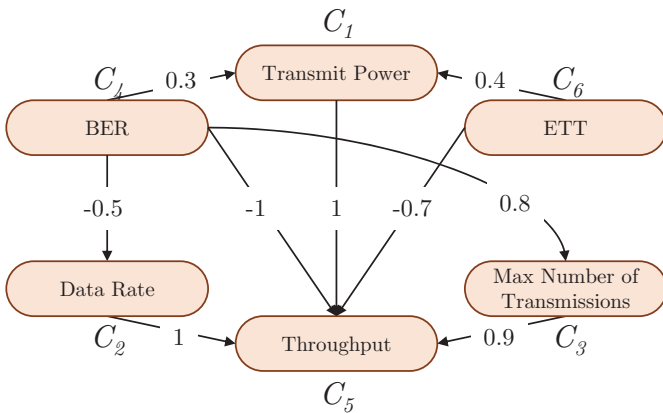


Fig. 22. WCM representing six processes within a wireless node.

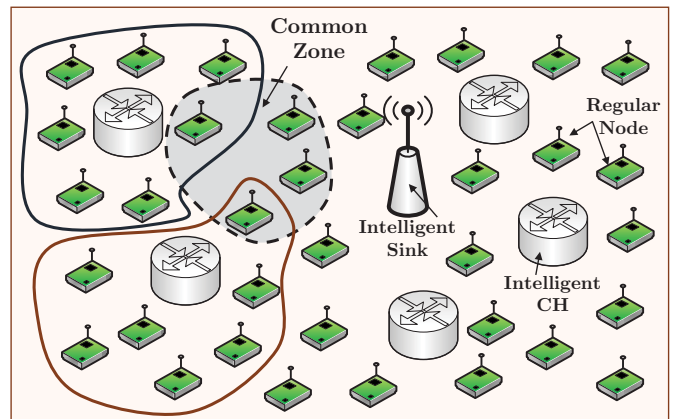


Fig. 23. Network topology with random deployment of nodes. The WCM is implemented in a distributed way on the CHs which act as intelligent nodes.

consumption of the network satisfies

$$\sum_{i=1}^{N_C} EnC_i < \frac{T \cdot N_{WSN} \cdot E}{X}, \quad (17)$$

mAhr per T time slots. Note that monitoring at the sink node takes place every T time slots. On the other hand, the end-to-end throughput can be monitored at the sink node by counting the number of packets successfully received per time slot. Thus, if R_{WSN} is the throughput required by the user, the task of the WCM at the sink is to make sure that the number of packets successfully received at the sink, ρ_S , in T time slots satisfies

$$\rho_S \geq T \cdot R_{WSN}. \quad (18)$$

Similarly, since N_C/N_{WSN} is the ratio of the number of nodes within a cluster to the total number of nodes, the WCM at the CH can calculate its throughput by counting the number of successfully received packets per time slot and making sure that the number of packets successfully received at the CH, ρ_{CH} , in $T/4$ time slots satisfies

$$\rho_{CH} \geq \frac{T \cdot N_C \cdot R_{WSN}}{4N_{WSN}}. \quad (19)$$

Thus, the designed WCM considers EnC and R_{WSN} as the end-to-end goals. It considers transmit power, data rate, duty cycle control, routing, source loading rate control, and scheduling active nodes as tools for achieving these goals. The environment variables that trigger the WCM to operate are PLR, channel utilization, buffer capacity, and node failure. Fig. 24 shows the overall WCM that is installed at the CHs and sink node.

The dashed edges in Fig. 24 are the ones going in/out of the goals of the WCM. Thus, every T/η time slots the WCM checks if any environment variable or goal violate their predefined threshold in order to take appropriate actions. For example, if cluster energy consumption is too high, the WCM has the option to decrease transmit power, decrease the number of active nodes, decrease the source loading rates, or invoke routing to find more energy efficient paths. This action is determined based on the current status of the network. For instance, if energy consumption is too high and the throughput is too low, then the WCM will not decrease the number of active nodes or loading rate because this will decrease throughput further. Instead routing may be invoked and transmit power reduced. This way, the WCM is able to consider multiple conflicting issues and deal with several challenges simultaneously.

2) *Performance analysis*: In order to study the performance of the WCM system, extensive computer simulations are performed. The WCM system is compared to TABU-RCC [140], which is a prominent network planning protocol, and Symphony [141], which is an energy-efficient transmit power and data rate control protocol. Different mobility scenarios are considered in order to examine the performance of the WCM system in dynamic environments. In the simulation set-up, sensor nodes have an average speed of 1.5 m/s and the network performance is tested under different network sizes. The WSN is assumed to be running an application with target network

lifetime of at least 1500 time slots and target throughput of at least 50 kbps.

It can be seen from the simulation results in Fig. 25 that the WCM system is the only one capable of achieving both (conflicting) goals at most network sizes. This is because of the ability of the WCM system to find compromises between conflicting goals and consider multiple issues simultaneously. TABU-RCC achieves good network lifetime results at the expense of throughput, while Symphony achieves only good throughput results at larger network sizes.

The performance of the WCM can be further improved through RL. As mentioned earlier, RL can be implemented in a distributed way and requires limited computations. This section presents Q -learning [137] to illustrate how to design a distributed learning process. Here the learning process will observe the actions of the WCM system (i.e., the reconfigurations made) and assigns rewards to those reconfigurations that achieve the predefined goals.

3) *Learning*: Since the WCM engine is implemented at the CHs, the learning process is also implemented there. The Q -learning algorithm consists of: Policy, reward system, and value function. Thus, the state of any node will be represented using the combination $\{\text{transmit power } (T_x), \text{ data rate } (R_{WSN}), \text{ next hop index } (NI)\}$. These parameters are chosen because they can be controlled by the WCM and can have significant impact on the performance of several nodes in the network. This means that the set of actions (or reconfigurations) observed by the learning protocol are $\{\text{increase transmit power, decrease transmit power, increase rate, decrease rate, change next hop}\}$. The action *change the next hop* occurs when routing is invoked.

To determine how the rewards are assigned, the application requirements are used to specify thresholds that can be monitored by the learning process. Therefore, if the reconfigurations made by the WCM lead to values of the aforementioned goals that are within the thresholds, a reward will be given. Otherwise the reward is zero. We specify an additional constraint that the thresholds of all goals have to be satisfied in order

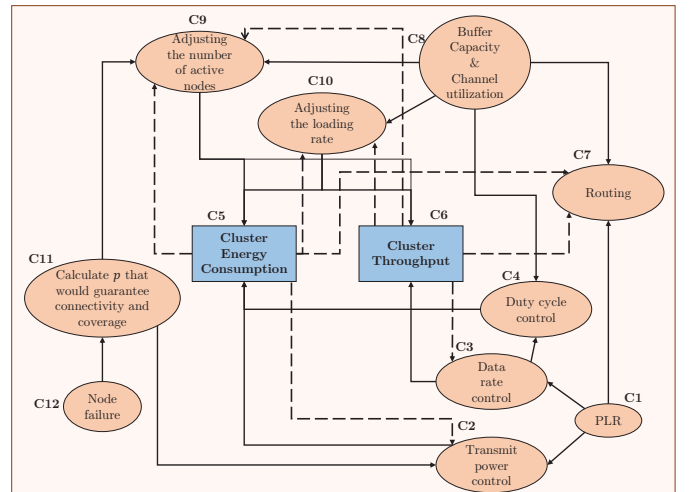


Fig. 24. Internal structure of the WCM engine as implemented in the cluster heads.

for the rewards to be assigned. If any goal is violated, the total reward is zero. This is to avoid accumulating rewards at actions that only achieve a subset of goals.

To implement distributed learning, DVF is used to combine knowledge from different CHs. Using DVF, the Q-values for each {state, action} pair is calculated as

$$Q(S_t, a_t) = r_{t+1} + \sigma \sum_{j \in \text{common_zone}} \omega_Q \times V_t^j(S_{t+1}) \quad (20)$$

where

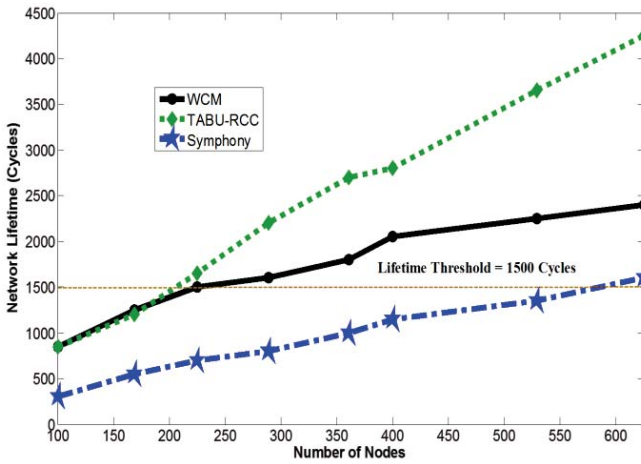
$$V_t^j(S_{t+1}) = \max_{a_{t+1}} Q(S_{t+1}, a_{t+1}) \quad (21)$$

and ω_Q is a weight factor that can be used to give more importance to information from certain nodes. In this example, equal importance is given to all nodes. Thus, $\omega_Q = 1/|\text{common_zone}|$, where $|\text{common_zone}|$ is the number of nodes in the common zone. However, in order to limit the overhead incurred from exchanging values from the Q-table, only the values concerning the nodes on the boundaries of each cluster are exchanged. This is because these boundary

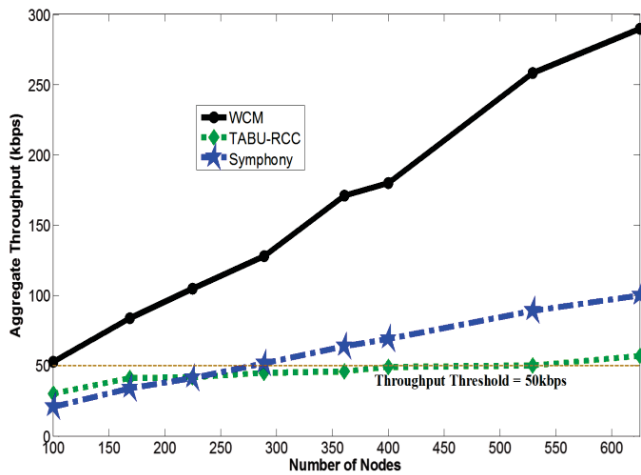
nodes are the ones expected to interfere with the performance of other nodes in neighboring clusters as illustrated in Fig. 23.

In the simulation results shown in Fig. 26, the mobility of the nodes is fixed at 1.5 m/s. In addition, the learning protocol is compared to the energy-efficient coverage and connectivity routing (EECCR) protocol [142], as well as TABU-RCC, and Symphony. Moreover, an independent learning protocol is simulated, where CHs do not exchange values and do not use the DVF. This will give an idea about how the cooperation of CHs can enhance performance. This experiment considers an application with target network lifetime of at least 1500 cycles and target throughput of at least 50kbps, while in the second experiment assumes that the objective is to maximize network lifetime with no regard to throughput.

It can be seen that the learning protocol improves the performance of the WCM system. This is because of the ability of the learning protocol to accumulate knowledge that allows the

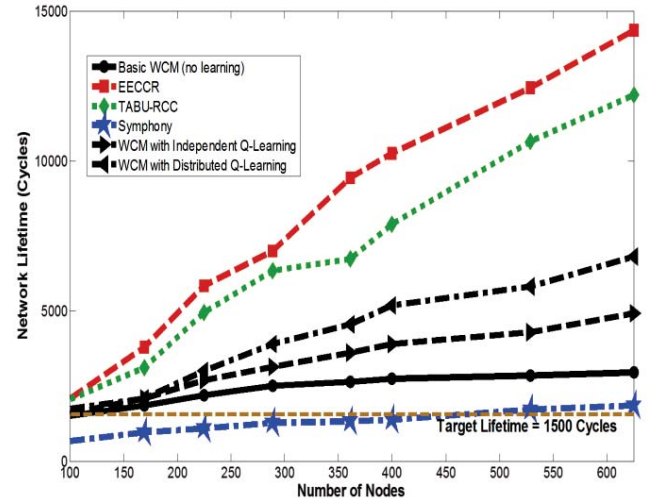


(a)

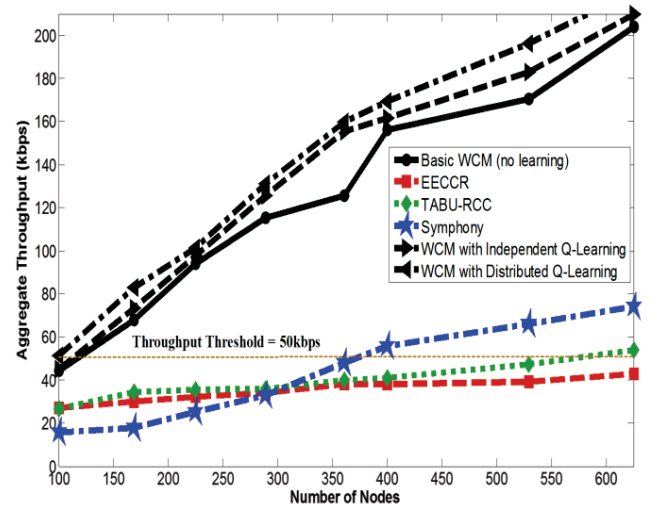


(b)

Fig. 25. Simulation results of WCM: (a) Network lifetime; (b) Network throughput.



(a)



(b)

Fig. 26. Simulation results of the learning protocol: (a) Network lifetime; (b) Network throughput.

WCM system to make choices for the reconfiguration actions to take place. Without the learning protocol, these choices are made blindly. In addition, distributed learning achieves better performance than independent learning, illustrating the importance of CH cooperation. Furthermore, the performance improvement is higher in larger networks. This is because larger networks have more options for the WCM to choose from.

4) *Complexity*: The WCM system imposes additional computational complexity and communication overhead on the regular WSN. However, it is important to notice that there are no loops in the execution of WCM algorithms. In addition, the WCM is an event-based system and the algorithms are executed upon detecting a significant change in one of the environment variables such as PLR or buffer capacity. Algorithms in other protocols that we have been using for comparison such as TABU-RCC or EECCR are executed periodically. This means that there is a chance they execute without there being a real need (e.g., when there are no changes in the network). With regards to communication overhead, the WCM system piggybacks control information on data packets. Special packets may be transmitted only if required, for example in the event of node failure, where immediate action is needed. This minimizes overhead and ensures robustness without incurring the risk of missing or delaying actions. Furthermore, the WCM system does not require centralized implementation. As we have seen, it can be implemented in CHs, with each CH controlling its local area. Note that TABU-RCC and EECCR protocols have to be implemented at the centralized sink node. Centralized solutions have to collect network-wide data and thus may incur high delays and communication overhead. Moreover, WCM is an entirely software-based network reconfiguration solution. Thus, implementing it does not require dedicated nodes or hardware modifications. To conclude, even though the WCM system imposes additional complexity, like any network management system, this extra complexity is minimum and justified by the significant improvement in performance.

C. Context-Awareness (CA)

CA is a classical research area in pervasive computing that has recently been adopted in mobile and wireless networking [15], [143]. Technically speaking, CA can be a part of any cognitive networking or cross-layer reconfiguration framework, which is why we think it deserves a section on its own. In addition, the growth of heterogeneous and ubiquitous networking makes CA quite important. CA targets the minimization of uncertainty in decision making by providing necessary information [15]. This improves the efficiency, confidence, and user experience in the application. CA is basically a collection of measured and inferred knowledge that arises from the general activity of a system and can exist independently of any system interactions [15].

There are four main types of uncertain context information that can appear in wireless networks [15]: Imperfect, ambiguous, wrong, or unknown context information. Imperfect context information can be used directly or can be used to infer

other information. It can arise from inaccurate, incomplete, or outdated context information. Unreliable communication links can be a cause of that. For example, in high mobility situations such as VANETs, links may break frequently causing context information to be missed. Eventually, information within one node may become obsolete. The second type of uncertain context information is ambiguous information. This can be the outcome of operating in heterogeneous environments where conflicting policies exist. For example, one node may decide to act selfishly to maximize its own throughput, thus causing a negative impact on the overall network performance. The third type of uncertain context information is wrong information, resulting from erroneous, inadequate, or inapplicable information. This can result from delivering information to the wrong entity or delivering incomplete information due to channel unreliability. Finally, unknown context information is generally expected information that has not been delivered. This may result from failures or node isolations.

CA systems try to bridge some of these information uncertainties. For example, all three cognitive standards discussed in the previous section include different forms of CA [112]. RRS includes specifications for acquisition of long-term and short-term context information. On the long-term the network collects information such as network status and spectrum availability, while on the short-term the information includes channel conditions, neighbor nodes, QoS parameters, and other information that may be necessary for JRRM. Similarly, IEEE 1900.4 collects context information about available spectrum bands, policies and regulations, and status of RANs. In addition, SON has an extensive CA framework that addresses different cases. For example, to minimize inter-cell interference, eNodeBs exchange information about interference indicators and transmit power used in the uplink and downlink directions. In addition, eNodeBs may exchange information about their current traffic load in order to perform load balancing.

The importance and potential of CA has been recognized by many researchers. Thus, in [144] the mobility management protocol known as Tramcar is proposed, which uses context information to make vertical handover decisions. This protocol considered context information about the cost of wireless services, power consumption, resource availability, network conditions, and security issues. At the application layer, user preferences are first translated into networking parameters. Then, a handoff necessity estimator uses context information about location, time, and places of events to avoid frequent handovers. In addition, the transport layer detect the presence of other networks and implement the decisions of the protocols at the application layer.

With regards to mobility management, the work in [78] has defined twelve necessary requirements for a context-aware system. These are: The availability of reliable information from all layers, standardized network discovery (particularly based on the IEEE 802.21 Media Independent Handover system [136]), post-handover self-management, consideration of the application profile, use of a single radio interface, pre-authentication (before handovers are executed), the consideration of multiple available networks, use of vendor independent handoff trigger indicators such as the number of packet re-

transmissions, it must take into account quality-of-experience (QoE) factors (such as costs and user preferences) as well as QoS factors, it must work with several protocols, admission control, and finally making the actual handover decision after considering all the parameters. Thus, the work in [78] provides a framework for any researcher wishing to propose context-aware mobility management systems. Other works include the framework for context management for next generation networks [138], the framework for context mediation proposed in [139], and the framework for collaborative CA proposed in [140].

CA has been utilized in ad hoc networks as well. However, CA systems in ad hoc networks usually take a narrower perspective (such as focusing on one particular issue or a small subset of objectives) due to the lack of a centralized node. Location awareness in particular has found extensive use in different types of ad hoc networks such as in VANETs, where it has been used for issues such as routing [141], QoS support [142], and security [145]. Mobility awareness has also been utilized for routing in mobile ad hoc networks [146]. In addition, CA has been used for data fusion in WSNs [147] to improve the efficiency of the process.

V. CONCLUSIONS AND OPEN RESEARCH TOPICS

The demand for sophisticated and ubiquitous applications has shifted the paradigm of wireless networks to reconfigurable wireless networks. Nevertheless, the scattered research in this area hinders the gains that can be achieved by reconfiguration. This paper presented an overview of reconfigurable wireless networks with an in-depth discussion of reconfiguration at all layers of the protocol stack. Reconfigurable networks have been classified in the paper into three levels: Reconfiguration at the physical layer, reconfigurable networking, and network intelligence. The paper focused on reconfigurable systems in emerging areas such as cognitive radio networks, cross-layer reconfiguration and software-defined networks. Moreover, the paper covered context awareness and cognitive networking as key enablers of reconfiguration in highly complex and dynamic networks. Furthermore, three case studies have been detailed, one for each level of reconfigurable networks. The first targeted management of CRNs where centralized and distributed network-based reconfigurations were used. The second was dedicated to cross-layer reconfiguration in M2M networks based on game theory. The third case study covered network intelligence in WSNs where a reasoning and learning framework was established to enable effective reconfiguration in WSNs. Several examples and experimental results have been presented in the paper illustrating the superiority of reconfigurable networks over non-reconfigurable ones, especially in the case of highly complex and dynamic networks. These results show that the reconfigurable wireless network paradigm will be a key enabler for next-generation wireless communications networks and services.

Although research in reconfigurable wireless networks has been quite extensive, there remain areas that need more attention. In this section, some of these areas are briefly discussed.

1) Learning: This is quite a powerful tool with great potential, which is why it has been recommended by standards such as RRS. Nevertheless, work in this area still faces challenges. One issue is the learning speed. The learning system has to observe the environment in order to accumulate knowledge. The time this takes depends on the parameters being considered. In dynamic environments, the parameters may be changing too frequently, rendering the learning system useless. To address this challenge, researchers typically confine the focus of the learning system to only a few parameters that can guarantee convergence. Another solution is to implement learning in a distributed way, across multiple agents. In this case, each agent collects information only from a subset of its neighbors. This may lead to suboptimal solutions and requires collecting agents that can aggregate the learned information. It also may require asynchronous operation, where each agent executes its learning algorithm according to the parameters in its neighborhood. In addition, the performance of learning systems in noisy environments, interference, changing topologies, or node failures needs investigation. In these situations, the learning system may become unstable and may have a negative effect on network performance. Finally, combining learning tools with CA approaches can have great potentials. Learning algorithms can be adapted to choose the most important parameters, given the right context, while CA systems can get more information about the history of the data being considered.

2) Coexistence and Interoperability: The techniques discussed in this paper provide solutions for a wide variety of challenges. However, there is limited work on the performance of reconfigurable systems under the coexistence of multiple solutions. For example, a learning system that targets the minimization of energy consumption may guide nodes to reduce their transmit powers. This may have an impact on packet loss, interference, and routing. Thus, an independent routing solution that is not considered in the learning process may suffer negatively. Interoperability is also another challenge. The world of wireless networks now includes a variety of standards, protocols, vendors, operators, etc. These are not always compatible. SDN is a promising solution to address this challenge, as explained above. However, SDN requires hardware modifications that may be expensive. The alternative direction that is currently pursued is to build increasingly complex reconfigurable devices capable of dealing with the incompatibilities of the different network components.

3) Intelligence in Reconfigurable CRNs: Most of the research in CRNs still focuses on the PHY layer. As indicated in this paper, tangible performance gains can be achieved if network-based reconfiguration is applied. Such reconfiguration can be achieved globally via an FC (in centralized CRNs), or locally via learning and consensus-based schemes (in decentralized CRNs). For multiband sensing, the existing literature has been limited to homogenous multiband detectors (which consist of single-type sensors), and thus reconfiguration has been restricted to adapting the sensors' parameters. Nevertheless, as shown in this paper, further gains can be accomplished if more than one sensor type are available at the SU device. In particular, it has been shown that the RMD outperforms

the well-known MJD in terms of the aggregate throughput of the CRN. This must motivate the research community to further investigate low-cost and reconfigurable heterogeneous multiband detectors.

4) *Context-Awareness*: There are still open issues in wireless network reconfiguration based on CA. One important challenge is the tradeoff in considering higher level versus lower level context information. On one hand, lower level context information enables fine-grained adaptation of network parameters. However, the parameters available in this level are numerous and the effect of considering only a subset of parameters must be considered. On the other hand, higher level information has a better chance to directly address the QoE of the user, but may not lead to fine adaptations. A promising direction that needs further investigation is context sharing. In a cooperative network, exchanging context information may lead to the improvement of all cooperating nodes. However, this consumes bandwidth. Thus, deciding which information is worth sharing with which node is important. There is also a need for researching various reasoning and learning techniques, and how they may coexist especially in complex dynamic networks.

5) *Security and Privacy*: Research in this area is usually performed independently of research in other branches in networking. In today's ubiquitous networking world, security has to be considered at the design stage, not as an afterthought. For example, CA requires gathering information about particular parameters. This may potentially violate certain privacy issues such as location, which is quite important in many networks such as VANETs. In addition, reconfiguration in heterogeneous environments has potential security risks. Not all networks have the same security guarantees. Thus, choosing a particular wireless service may introduce vulnerabilities. If the violated node includes critical information about another network (such as an SDN controller managing multiple networks), then the vulnerability will stretch across multiple networks. Thus, trust management and security have to be an integral part of the reconfiguration process.

REFERENCES

- [1] M. Mills, "The cloud begins with coal: Big data, big networks, big infrastructure, and big power," Digital Power Group, Tech. Rep., 2013.
- [2] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2013-2018, Cisco Systems, Inc., Feb. 2014.
- [3] Ericsson Mobility Report, Ericsson, 2013.
- [4] M. Ibnkahla, *Adaptation and Cross-layer Design in Wireless Networks*. Boca Raton, FL, USA: CRC Press, 2008.
- [5] H. Arslan, "Adaptation techniques and the enabling parameter estimation algorithms for wireless communications," in *Signal Processing for Mobile Communications Handbook*, M. Ibnkahla, Ed. CRC Press, Jul. 2004, ch. 22.
- [6] M. Ibnkahla, *Adaptive Signal Processing for Wireless Communications*, ser. Adaptation in Wireless Networks. Boca Raton, FL, USA: CRC Press, 2008.
- [7] B. Clerckx and C. Oestges, *MIMO Wireless Networks: Channels, Techniques and Standards for Multi-Antenna, Multi-User and Multi-Cell Systems*, 2nd ed. Academic Press, 2013.
- [8] S. Sun, Q. Gao, Y. Peng, Y. Wang, and L. Song, "Interference management through CoMP in 3GPP LTE-advanced networks," *IEEE Wireless Commun. Mag.*, vol. 20, no. 1, pp. 59–66, Feb. 2013.
- [9] M. Sternad, T. Svensson, T. Ottosson, A. Ahlen, A. Svensson, and A. Brunstrom, "Towards systems beyond 3G based on adaptive OFDMA transmission," *Proc. IEEE*, vol. 95, no. 12, pp. 2432–2455, Dec. 2007.
- [10] J. Mietzner, R. Schober, L. Lampe, W. Gerstacker, and P. Hoeher, "Multiple-antenna techniques for wireless communications - a comprehensive literature survey," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 2, pp. 87–105, Second 2009.
- [11] N. Miridakis and D. Vergados, "A survey on the successive interference cancellation performance for single-antenna and multiple-antenna OFDM systems," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 312–335, First 2013.
- [12] T. Ulversoy, "Software defined radio: Challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 4, pp. 531–550, Fourth 2010.
- [13] R. Edirisinghe and A. Zaslavsky, "Cross-layer contextual interactions in wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 1114–1134, Second 2014.
- [14] S. Haykin, M. Fatemi, P. Setoodeh, and Y. Xue, "Cognitive control," *Proc. IEEE*, vol. 100, no. 12, pp. 3156–3169, Dec. 2012.
- [15] P. Makris, D. Skoutas, and C. Skianis, "A survey on context-aware mobile and wireless networking: On networking and computing environments' integration," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 362–386, First 2013.
- [16] H. Al-Asady and M. Ibnkahla, "Adaptive modulation over nonlinear time-varying channels," *European Transactions on Communications*, vol. 18, pp. 685–692, 2007.
- [17] K. Arshad, R. Mackenzie, U. Celentano, A. Drozdy, S. Leveil, G. Mange, J. Rico, A. Medela, and C. Rosik, "Resource management for QoS support in cognitive radio networks," *IEEE Commun. Mag.*, vol. 52, no. 3, pp. 114–120, Mar. 2014.
- [18] I. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 40–48, Apr. 2008.
- [19] G. Hattab and M. Ibnkahla, "Multiband spectrum access: Great promises for future cognitive radio networks," *Proc. IEEE*, vol. 102, no. 3, pp. 282–306, Mar. 2014.
- [20] K. Chowdhury, M. Di Felice, and I. Akyildiz, "TCP CRAHN: A transport control protocol for cognitive radio ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 4, pp. 790–803, Apr. 2013.
- [21] P. Hu, "Cognitive radio ad hoc networks: A local control approach," Ph.D. dissertation, Dept. Elect. and Comput. Eng., Queen's Univ., Feb. 2013.
- [22] Y.-C. Liang, K.-C. Chen, G. Li, and P. Mahonen, "Cognitive radio networking and communications: an overview," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 3386–3407, Sep. 2011.
- [23] M. Ibnkahla, *Cooperative Cognitive Radio Networks: The Complete Spectrum Cycle*. CRC Press- Taylor & Francis, 2014.
- [24] Y.-C. Liang, Y. Zeng, E. Peh, and A. T. Hoang, "Sensing-throughput tradeoff for cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1326–1337, Apr. 2008.
- [25] E. Axell, G. Leus, E. Larsson, and H. Poor, "Spectrum sensing for cognitive radio : State-of-the-art and recent advances," *IEEE Signal Process. Mag.*, vol. 29, no. 3, pp. 101–116, May 2012.
- [26] B. Farhang-Boroujeny, "Filter bank spectrum sensing for cognitive radios," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1801–1811, May 2008.
- [27] J. Heo, H. Ju, S. Park, E. Kim, and D. Hong, "Simultaneous sensing and transmission in cognitive radio," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 1948–1959, Apr. 2014.
- [28] J. Ma, G. Li, and B.-H. Juang, "Signal processing in cognitive radio," *Proc. IEEE*, vol. 97, no. 5, pp. 805–823, May 2009.
- [29] C.-H. Liu, J. Tran, P. Pawelczak, and D. Cabric, "Traffic-aware channel sensing order in dynamic spectrum access networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 11, pp. 2312–2323, Nov. 2013.
- [30] G. Hattab and M. Ibnkahla, "Enhanced pilot-based spectrum sensing algorithm," in *Proc. IEEE Biennial Symp. on Commun. (QBSC)'14*, Jun. 2014, pp. 57–60.
- [31] —, "Improved pilot-tone aided detection for cognitive radio applications," *IEEE Trans. Signal Process.*, Submitted For Publication.
- [32] E. Axell and E. Larsson, "Optimal and sub-optimal spectrum sensing of OFDM signals in known and unknown noise variance," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 2, pp. 290–304, Feb. 2011.
- [33] Z. Quan, S. Cui, A. Sayed, and H. Poor, "Optimal multiband joint detection for spectrum sensing in cognitive radio networks," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1128–1140, Mar. 2009.
- [34] P. Paysarvi-Hoseini and N. Beaulieu, "Optimal wideband spectrum sensing framework for cognitive radio systems," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1170–1182, Mar. 2011.
- [35] —, "On the benefits of multichannel/wideband spectrum sensing with non-uniform channel sensing durations for cognitive radio net-

- works," *IEEE Trans. Commun.*, vol. 60, no. 9, pp. 2434–2443, Sep. 2012.
- [36] S. Chaudhari, J. Lunden, V. Koivunen, and H. Poor, "Cooperative sensing with imperfect reporting channels: Hard decisions or soft decisions?" *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 18–28, Jan. 2012.
- [37] Z. Li, F. Yu, and M. Huang, "A distributed consensus-based cooperative spectrum-sensing scheme in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 383–393, Jan. 2010.
- [38] P. Hu and M. Ibnkahla, "A consensus-based protocol for spectrum sharing fairness in cognitive radio ad hoc and sensor networks," *Int. J. of Distributed Sensor Networks*, vol. 2012, no. 11, pp. 1–12, Aug. 2012.
- [39] —, "Consensus-based local control schemes for spectrum sharing in cognitive radio sensor networks," in *Proc. 26th Biennial Symp. on Commun. (QBSC'12)*, May 2012, pp. 115–118.
- [40] W. Ejaz, N. ul Hasan, and H. S. Kim, "Distributed cooperative spectrum sensing in cognitive radio for ad hoc networks," *Computer Communications*, pp. 1341 – 1349, 2013.
- [41] A. Sayed, "Adaptive networks," *Proc. IEEE*, vol. 102, no. 4, pp. 460–497, Apr. 2014.
- [42] K. Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proc. IEEE*, vol. 97, no. 5, pp. 878–893, May 2009.
- [43] B. Fu, Y. Xiao, H. Deng, and H. Zeng, "A survey of cross-layer designs in wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 110–126, First 2014.
- [44] C. Cox, *An Introduction to LTE: LTE, LTE-Advanced, SAE and 4G Mobile Communications*. Wiley, 2012.
- [45] A. Taha, N. Ali, and H. Hassanein, *LTE, LTE-Advanced and WiMAX: Towards IMT-Advanced Networks*, ser. ITPro collection. Wiley, 2011.
- [46] D. Lee, H. Seo, B. Clerckx, E. Hardouin, D. Mazzarese, S. Nagata, and K. Sayana, "Coordinated multipoint transmission and reception in LTE-advanced: deployment scenarios and operational challenges," *IEEE Commun. Mag.*, vol. 50, no. 2, pp. 148–155, Feb. 2012.
- [47] E. Pateromichelakis, M. Shariat, A. ul Qudus, and R. Tafazolli, "On the evolution of multi-cell scheduling in 3GPP LTE / LTE-A," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 701–717, Second 2013.
- [48] K. Pedersen, T. Kolding, F. Frederiksen, I. Kovacs, D. Laselva, and P. Mogensen, "An overview of downlink radio resource management for UTRAN long-term evolution," *IEEE Commun. Mag.*, vol. 47, no. 7, pp. 86–93, Jul. 2009.
- [49] N. Prasad, H. Zhang, H. Zhu, and S. Rangarajan, "Multi-user MIMO scheduling in the fourth generation cellular uplink," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4272–4285, Sep. 2013.
- [50] G. Araniti, M. Condoluci, L. Militano, and A. Iera, "Adaptive resource allocation to multicast services in lte systems," *IEEE Trans. Broadcast.*, vol. 59, no. 4, pp. 658–664, Dec. 2013.
- [51] B.-J. Chang, Y.-H. Liang, and T.-S. Tsai, "Cross-layer channel selection and reward-based power allocation for increasing system capacity and reward in multiple-input-multiple-output wireless communications," *Communications, IET*, vol. 7, no. 10, pp. 1032–1041, Jul. 2013.
- [52] M. Cheng, X. Gong, L. Cai, and X. Jia, "Cross-layer throughput optimization with power control in sensor networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 3300–3308, Sep. 2011.
- [53] H. Ammari and S. Das, "Centralized and clustered k-coverage protocols for wireless sensor networks," *IEEE Trans. Comput.*, vol. 61, no. 1, pp. 118–133, Jan. 2012.
- [54] S.-H. Cha, K.-W. Lee, , and H.-S. Cho, "Predictive grid-based predictive geographical routing for inter-vehicle communication in urban areas," *Int. Journal of Distributed Sensor Networks*, vol. 2012, pp. 1–7, 2012.
- [55] M.-W. Ryu, S.-H. Cha, J.-G. Koh, S. Kang, and K.-H. Cho, "Position-based routing algorithm for improving reliability of inter-vehicle communication," *KSII Trans. on Internet and Information Systems*, vol. 5, Aug. 2011.
- [56] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Wireless Commun. Mag.*, vol. 46, no. 9, pp. 59–67, Sep. 2008.
- [57] C.-X. Wang, F. Haider, X. Gao, X.-H. You, Y. Yang, D. Yuan, H. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.
- [58] D. Boswarthick, O. Elloumi, and O. Hersent, *M2M Communications: A systems approach*. John Wiley & Sons, 2012.
- [59] A. Lo, Y. Law, and M. Jacobsson, "A cellular-centric service architecture for machine-to-machine (M2M) communications," *IEEE Wireless Commun. Mag.*, vol. 20, no. 5, pp. 143–151, Oct. 2013.
- [60] *Study on RAN improvements for machine-type communications*, 3GPP TR 37.868, 2012.
- [61] "Air interface for broadband wireless access systems- amendment 1: Enhancements to support machine-to-machine applications," *IEEE 802.16p*, 2012.
- [62] A. Bleicher, "The 5G phone future: Samsung's millimeter-wave transceiver technology could enable ultrafast mobile broadband by 2020," *IEEE Spectrum*, vol. 50, no. 7, pp. 15–16, 2013.
- [63] F. Rusek, D. Persson, B. K. Lau, E. Larsson, T. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.
- [64] D. Monderer and L. Shapley, "Potential games," *Games and economic behavior*, vol. 14, no. 1, pp. 124–143, 1996.
- [65] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Networks*, vol. 9, no. 3, pp. 228–248, 2011.
- [66] M. Youssef, M. Ibrahim, M. Abdelatif, L. Chen, and A. Vasilakos, "Routing metrics of cognitive radio networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 92–109, First 2014.
- [67] W. Sandholm, "Potential games with continuous player sets," *Journal of Economic Theory*, vol. 97, no. 1, pp. 81–108, 2001.
- [68] *Further advancements for E-UTRA physical layer aspects (Release 9)*, 3GPP TS, 2010.
- [69] S. Rangan, T. Rappaport, and E. Erkip, "Millimeter-wave cellular wireless networks: Potentials and challenges," *Proc. IEEE*, vol. 102, no. 3, pp. 366–385, Mar. 2014.
- [70] T. Bai and R. W. H. Jr., "Coverage and rate analysis for millimeter wave cellular networks," 2014. [Online]. Available: <http://arxiv.org/abs/1402.6430>
- [71] C. W. Pyo and M. Hasegawa, "Minimum weight routing based on a common link control radio for cognitive wireless ad hoc networks," in *Proc. Int. Conf. on Wireless Commun. and Mobile Computing*, 2007, pp. 399–404.
- [72] Z. Yang, G. Cheng, W. Liu, W. Yuan, and W. Cheng, "Local coordination based routing and spectrum assignment in multi-hop cognitive radio networks," *Mobile Networks and Applications*, vol. 13, no. 1-2, pp. 67–81, 2008.
- [73] A. Sampath, L. Yang, L. Cao, H. Zheng, and B. Y. Zhao, "High throughput spectrum-aware routing for cognitive radio networks," in *Proc. IEEE Int. Conf. on Cognitive Radio Oriented Wireless Networks and Commun. (CROWNCOM'07)*, 2007.
- [74] A. Abbagnale and F. Cuomo, "Gymkhana: A connectivity-based routing scheme for cognitive radio ad hoc networks," in *Proc. IEEE Conf. on Computer Commun. Workshops (INFOCOM'10)*, Mar. 2010, pp. 1–5.
- [75] I. Mustika, K. Yamamoto, H. Murata, and S. Yoshida, "Potential game approach for self-organization scheme in open access heterogeneous networks," in *Proc. IEEE Int. Conf. on Cognitive Radio Oriented Wireless Networks and Commun. (CROWNCOM'11)*, June 2011, pp. 216–220.
- [76] M. Li, Z. Li, and A. Vasilakos, "A survey on topology control in wireless sensor networks: Taxonomy, comparative study, and open issues," *Proc. IEEE*, vol. 101, no. 12, pp. 2538–2557, Dec. 2013.
- [77] A. Aziz, Y. Sekercioglu, P. Fitzpatrick, and M. Ivanovich, "A survey on distributed topology control techniques for extending the lifetime of battery powered wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 121–144, First 2013.
- [78] S. Fernandes and A. Karmouch, "Vertical mobility management architectures in wireless networks: A comprehensive survey and future directions," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 1, pp. 45–63, First 2012.
- [79] R.-H. Liou, Y.-B. Lin, and S.-C. Tsai, "An investigation on LTE mobility management," *IEEE Trans. Mobile Comput.*, vol. 12, no. 1, pp. 166–176, Jan 2013.
- [80] Y. Lee, T. Chuah, J. Loo, and A. Vinel, "Recent advances in radio resource management for heterogeneous LTE/LTE-A networks," *IEEE Commun. Surveys Tuts.*, no. 99, pp. 1–39, 2014.
- [81] "IEEE 802.11r 2008: Amendment 2: Fast basic service set (BSS) transition," IEEE, Tech. Rep., 2008.
- [82] "IEEE 802.11k-2008 - amendment 1: Radio resource measurement of wireless LANs," IEEE, Tech. Rep., 2008.
- [83] "IEEE 802.11s 2011: Amendment for mesh networking," IEEE, Tech. Rep., 2011.
- [84] Y. Fukuda, T. Abe, and Y. Oie, "Decentralized access point selection architecture for wireless LANs," in *Wireless Telecommun. Symp.*, 2004, May 2004, pp. 137–145.

- [85] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot, "Measurement-based self organization of interfering 802.11 wireless access networks," in *Proc. IEEE Int. Conf. on Computer Commun. (INFOCOM'07)*, May 2007, pp. 1451–1459.
- [86] Y. Zhu, Q. Ma, C. Bisdikian, and C. Ying, "User-centric management of wireless LANs," *IEEE Trans. Netw. Service Manag.*, vol. 8, no. 3, pp. 165–175, Sep. 2011.
- [87] L. Hanzo II and R. Tafazolli, "Admission control schemes for 802.11-based multi-hop mobile ad hoc networks: a survey," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 4, pp. 78–108, Fourth 2009.
- [88] V. Gabale, B. Raman, P. Dutta, S. Kalyanraman, B. Raman, P. Dutta, and S. Kalyanraman, "A classification framework for scheduling algorithms in wireless mesh networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 199–222, First 2013.
- [89] M. Kafi, D. Djenouri, J. Ben-Othman, and N. Badache, "Congestion control protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, no. 99, pp. 1–22, 2014.
- [90] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using openflow: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 493–512, First 2014.
- [91] B. Nunes, M. Mendonca, X. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *Communications Surveys Tutorials, IEEE*, no. 99, pp. 1–18, 2014.
- [92] F. Hu, Q. Hao, and K. Bao, "A survey on software defined networking (SDN) and openflow: From concept to implementation," *IEEE Commun. Surveys Tuts.*, no. 99, pp. 1–1, 2014.
- [93] [Online]. Available: <http://www.noxrepo.org/>
- [94] [Online]. Available: <https://openflow.stanford.edu/display/Beacon/Home>
- [95] [Online]. Available: <http://www.projectfloodlight.org/floodlight/>
- [96] A. Doria, J. Salim, R. Haas, H. Khosravi, W. Wang, L. Dong, R. Gopal, and J. Halpern, "Forwarding and control element separation (forces) protocol specification." [Online]. Available: <http://tools.ietf.org/html/rfc5810>
- [97] I. W. Group, "Interface to the routing system (IRS)." [Online]. Available: <https://datatracker.ietf.org/wg/irs/charter/>
- [98] O. N. Foundation. [Online]. Available: <https://www.opennetworking.org>
- [99] S. Sezer, S. Scott-Hayward, P. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, and N. Rao, "Are we ready for SDN? implementation challenges for software-defined networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 36–43, Jul. 2013.
- [100] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Trans. Commun.*, vol. 51, no. 2, pp. 114–119, Feb. 2013.
- [101] A. Martinez, M. Yannuzzi, V. Lopez, D. Lopez, W. Ramirez, R. Serral-Gracia, X. Masip-Bruin, M. Maciejewski, and J. Altmann, "Network management challenges and trends in multi-layer and multi-vendor settings for carrier-grade networks," *IEEE Commun. Surveys Tuts.*, no. 99, 2014.
- [102] D. Crisan, R. Birke, K. Barabash, R. Cohen, and M. Gusat, "Datacenter applications in virtualized networks: A cross-layer performance study," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 1, pp. 77–87, Jan. 2014.
- [103] J. Gruen, M. Karl, and T. Herfet, "Network supported congestion avoidance in software-defined networks," in *Proc. IEEE Int. Conf. on Networks (ICON'13)*, Dec. 2013, pp. 1–6.
- [104] Y. Hu, W. Wang, X. Gong, X. Que, and S. Cheng, "Balanceflow: Controller load balancing for openflow networks," in *Proc. IEEE Int. Conf. on Cloud Computing and Intelligent Syst. (CCIS'12)*, vol. 02, Oct. 2012, pp. 780–785.
- [105] S. Gutz, A. Story, C. Schlesinger, and N. Foster, "Splendid isolation: A slice abstraction for software-defined networks," in *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*. ACM, 2012, pp. 79–84.
- [106] S. Scott-Hayward, G. O'Callaghan, and S. Sezer, "SDN security: A survey," in *Proc. IEEE Int. Conf. on SDN for Future Networks and Services (SDN4FNS'13)*, Nov. 2013, pp. 1–7.
- [107] H. Egilmez, S. Dane, K. Bagci, and A. Tekalp, "OpenQoS: An openflow controller design for multimedia delivery with end-to-end quality of service over software-defined networks," in *Proc. IEEE Conf. and Summit on Signal Inform. Process. Assoc. Annu. (APSIPA ASC'12)*, Dec. 2012, pp. 1–8.
- [108] M. Bari, S. Chowdhury, R. Ahmed, and R. Boutaba, "Polycop: An autonomic QoS policy enforcement framework for software defined networks," in *Proc. IEEE Int. Conf. on SDN for Future Networks and Services (SDN4FNS'13)*, Nov 2013, pp. 1–7.
- [109] R. Thomas, D. Friend, L. DaSilva, and A. MacKenzie, "Cognitive networks," in *Cognitive Radio, Software-Defined Radio, and Adaptive Wireless Systems*. Springer, 2007, pp. 17–41.
- [110] G. Vijay, E. Ben Ali Bdira, and M. Ibnkahla, "Cognition in wireless sensor networks: A perspective," *IEEE Sensors J.*, vol. 11, no. 3, pp. 582–592, Mar. 2011.
- [111] J. Mitola and J. Maguire, G.Q., "Cognitive radio: making software radios more personal," *IEEE Personal Commun. Mag.*, vol. 6, no. 4, pp. 13–18, Aug. 1999.
- [112] K. Tsagkaris, P. Vlacheas, G. Athanasiou, V. Stavroulaki, S. Filin, H. Harada, J. Gebert, and M. Mueck, "Autonomics in wireless network management: Advances in standards and further challenges," *IEEE Netw.*, vol. 25, no. 6, pp. 41–49, Nov. 2011.
- [113] T. R. ETSI, "Reconfigurable radio systems (RRS): Functional architecture (FA) for the management and control of reconfigurable radio systems," T. R. ETSI, Tech. Rep., 2000.
- [114] "IEEE Standard for architectural building blocks enabling network-device distributed decision making for optimized radio resource usage in heterogeneous wireless access networks," *IEEE Std 1900.4-2009*, pp. C1–119, Feb. 2009.
- [115] *Technical Specification, 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunications Management; Self-Organizing Networks (SON); Self-healing Concepts and Requirements (Release 10)*, 3GPP TS 32.541, 2011.
- [116] H. Chen, C. Tse, and J. Feng, "Minimizing effective energy consumption in multi-cluster sensor networks for source extraction," *IEEE Trans. Wireless Commun.*, vol. 8, no. 3, pp. 1480–1489, Mar. 2009.
- [117] M. Bala Krishna and M. Doja, "Swarm intelligence-based topology maintenance protocol for wireless sensor networks," *Wireless Sensor Syst., IET*, vol. 1, no. 4, pp. 181–190, Dec. 2011.
- [118] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjørungnes, *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*, ser. Cambridge books online. Cambridge University Press, 2012.
- [119] Q. Wang, W. Yan, and Y. Shen, "N-person card game approach for solving set k-cover problem in wireless sensor networks," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 5, pp. 1522–1535, May 2012.
- [120] H. Ren and M.-H. Meng, "Game-theoretic modeling of joint topology control and power scheduling for wireless heterogeneous sensor networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 4, pp. 610–625, Oct. 2009.
- [121] D. Holmes and L. Jain, *Innovations in Bayesian Networks: Theory and Applications*. Springer Publications, 2008.
- [122] G. Chin, S. Choudhury, L. Kangas, S. McFarlane, and A. Marquez, "Fault detection in distributed climate sensor networks using dynamic bayesian networks," in *Proc. IEEE Int. Conf. on e-Science*, Dec. 2010, pp. 121–128.
- [123] C.-K. Lin and O. Kure, "Energy-aware path selection in mobile wireless sensor networks: A dynamic bayesian game approach," in *Proc. IEEE Int. Symp. on Personal, Indoor and Mobile Radio Commun.*, Sep. 2009, pp. 1198–1203.
- [124] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal, "Reinforcement learning for context-awareness and intelligence in wireless networks: Review, new features and open issues," *Journal of Network and Comput. Applicat.*, vol. 35, no. 1, pp. 253–267, 2011.
- [125] R. Sutton and A. Barto, *Reinforcement Learning: an Introduction*. Cambridge: The MIT Press, 1998.
- [126] E. Alpaydin, *Introduction to Machine Learning*. Massachusetts: The MIT Press, 2004.
- [127] A. Bhorkar, M. Naghshvar, T. Javidi, and B. Rao, "Adaptive opportunistic routing for wireless ad hoc networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 243–256, Feb. 2012.
- [128] G. Naddafzadeh-Shirazi, P.-Y. Kong, and C.-K. Tham, "Distributed reinforcement learning frameworks for cooperative retransmission in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 8, pp. 4157–4162, Oct. 2010.
- [129] T. Jiang, D. Grace, and Y. Liu, "Two-stage reinforcement-learning-based cognitive radio with exploration control," *Commun., IET*, vol. 5, no. 5, pp. 644–651, Mar. 2011.
- [130] K. Prabuchandran, S. K. Meena, and S. Bhatnagar, "Q-learning based energy management policies for a single sensor node with finite buffer," *IEEE Wireless Commun. Lett.*, vol. 2, no. 1, pp. 82–85, February 2013.
- [131] J. G. Schneider, W.-K. Wong, A. W. Moore, and M. A. Riedmiller, "Distributed value functions," in *Proc. of the Sixteenth Int. Conf. on Machine Learning*, 1999, pp. 371–378.

- [132] P. Ray and P. Varshney, "Estimation of spatially distributed processes in wireless sensor networks with random packet loss," *IEEE Trans. Wireless Commun.*, vol. 8, no. 6, pp. 3162–3171, Jun. 2009.
- [133] T. Clancy, A. Khawar, and T. Newman, "Robust signal classification using unsupervised learning," *IEEE Trans. Wireless Commun.*, vol. 10, no. 4, pp. 1289–1299, Apr. 2011.
- [134] Z. Han, R. Zheng, and H. Poor, "Repeated auctions with bayesian non-parametric learning for spectrum access in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 3, pp. 890–900, Mar. 2011.
- [135] J. Dickerson and B. Kosko, "Virtual worlds as fuzzy cognitive maps," in *Proc. IEEE Int. Symp. on Virtual Reality*, Sep. 1993, pp. 471–477.
- [136] "IEEE draft standard for local and metropolitan area networks – part 21: Media independent handover services," *IEEE P802.21c/D7*, December 2013, 2008.
- [137] A. El-Mougy and M. Ibnkahla, "A cognitive framework for WSN based on weighted cognitive maps and Q-learning," *Ad Hoc Networks*, vol. 16, pp. 46–69, 2014.
- [138] C. Baladron, J. Aguiar, B. Carro, L. Calavia, A. Cadenas, and A. Sanchez-Esguevillas, "Framework for intelligent service adaptation to user's context in next generation networks," *IEEE Commun. Mag.*, vol. 50, no. 3, pp. 18–25, Mar. 2012.
- [139] N. Roy, S. Das, and C. Julien, "Resource-optimized quality-assured ambiguous context mediation framework in pervasive environments," *IEEE Trans. Mobile Comput.*, vol. 11, no. 2, pp. 218–229, Feb. 2012.
- [140] C. Anagnostopoulos, S. Hadjiefthymiades, and E. Zervas, "Information dissemination between mobile nodes for collaborative context awareness," *IEEE Trans. Mobile Comput.*, vol. 10, no. 12, pp. 1710–1725, Dec. 2011.
- [141] M. Al-Rabayah and R. Malaney, "A new scalable hybrid routing protocol for VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2625–2635, Jul. 2012.
- [142] H. Saleet, R. Langar, K. Naik, R. Boutaba, A. Nayak, and N. Goel, "Intersection-based geographical routing protocol for VANETs: A proposal and analysis," *IEEE Trans. Veh. Technol.*, vol. 60, no. 9, pp. 4560–4574, Nov. 2011.
- [143] A. E. Mougy, A. Kamoun, M. Ibnkahla, S. Tazi, and K. Drira, "A context and application-aware framework for resource management in dynamic collaborative wireless M2M networks," *Journal of Network and Computer Applications*, vol. 44, pp. 30–45, 2014.
- [144] A. Hasswa, N. Nasser, and H. Hassanein, "A seamless context-aware architecture for fourth generation wireless networks," *Wireless Personal Communications*, vol. 43, no. 3, pp. 1035–1049, 2007.
- [145] R. Lu, X. Lin, X. Liang, and X. Shen, "A dynamic privacy-preserving key management scheme for location-based services in VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 127–139, Mar. 2012.
- [146] K. Manjappa and R. Guddeti, "Mobility aware-termite: a novel bio inspired routing protocol for mobile ad-hoc networks," *Networks, IET*, vol. 2, no. 4, pp. 188–195, Dec. 2013.
- [147] F. Ren, J. Zhang, Y. Wu, T. He, C. Chen, and C. Lin, "Attribute-aware data aggregation using potential-based dynamic routing in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 5, pp. 881–892, May 2013.